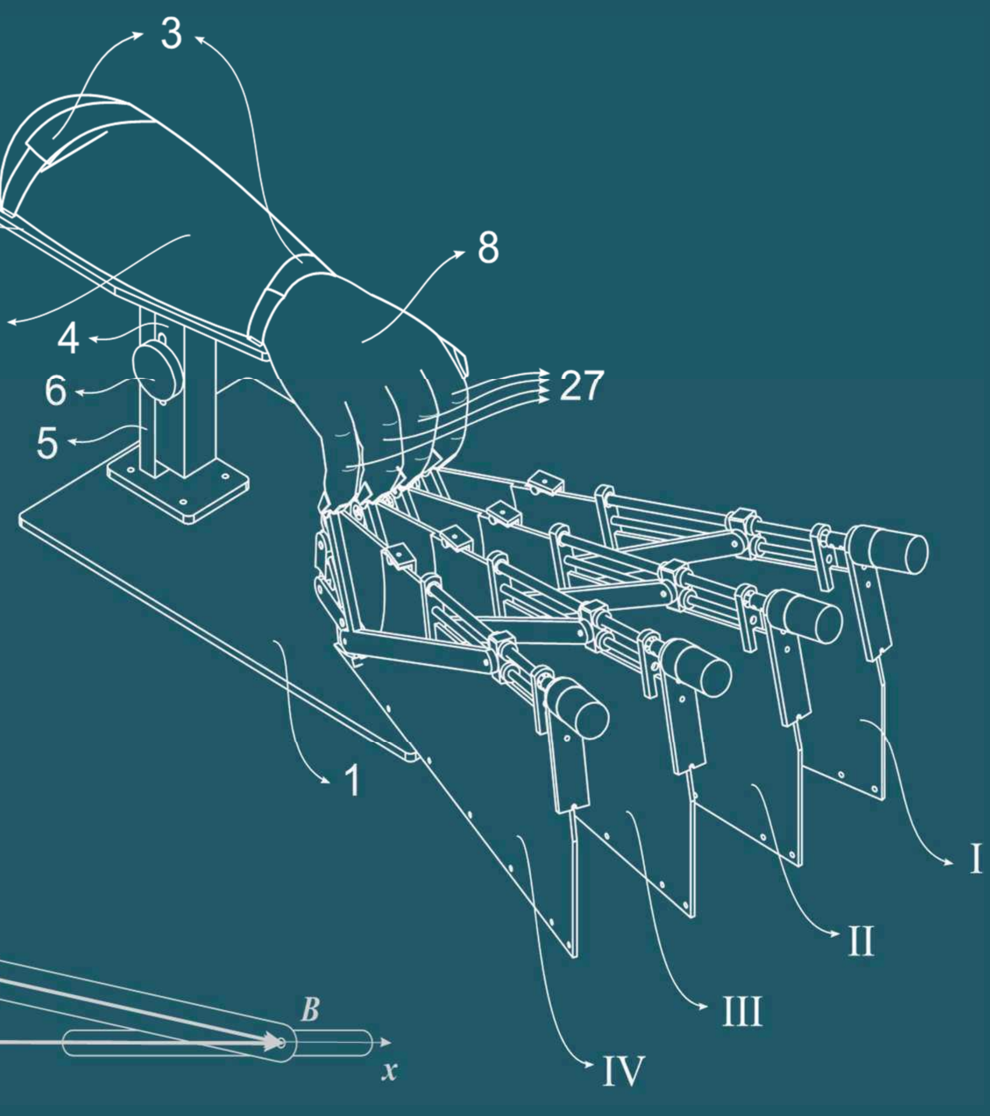
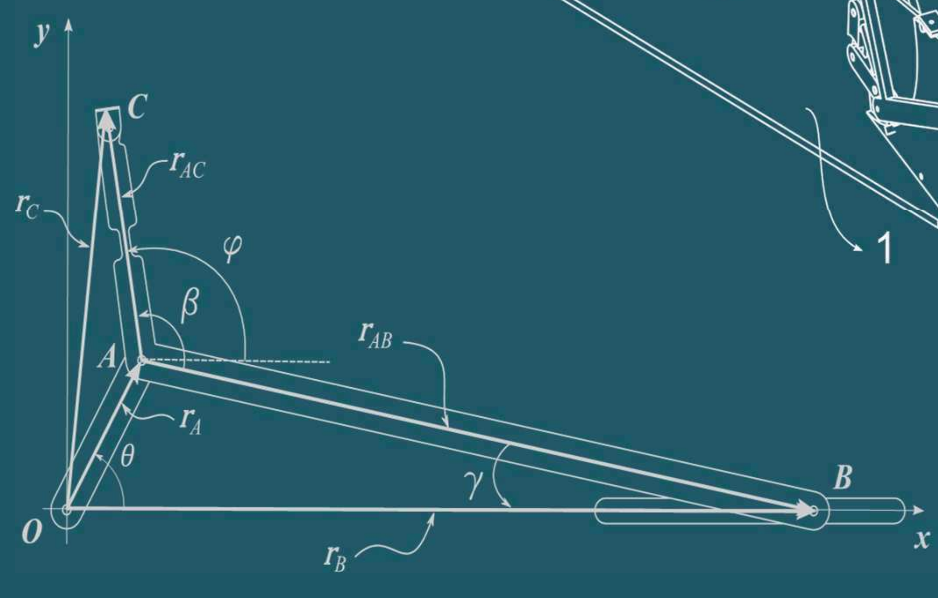
Mecatrónica y Robótica de Servicio: Teoría y Aplicaciones

E. Castillo Castañeda, P.A. Niño Suárez,
E. Morales Sánchez, J. E. Vargas Soto
y J.M. Ramos Arreguín





Asociación Mexicana de Mecatrónica A.C.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones.
Editado por Eduardo Castillo Castañeda, Paola Andrea Niño Suárez,
Eduardo Morales Sánchez, José Emilio Vargas Soto
y Juan Manuel Ramos Arreguín.

Publicado por
Asociación Mexicana de Mecatrónica, A. C.

Primera edición, Noviembre 2016
Impreso en Querétaro, México.

ISBN 978-607-9394-06-6

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Índice

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Índice

Índice

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 1, páginas 1-8.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

An estimation acoustic transducer based on sliding modes control

Julio César Tovar Rodríguez, Floriberto Ortíz Rodríguez, Carlos Román Mariaca Gaspar

***Abstract* — Electronic sensor devices in geophysical processes are required to measure and automate different tasks. Throughout history, people have created multiple type devices, but acoustics have an important application such as the content form description in deep wells, watersheds, lakes, caves, among others. The acoustic signal is capable of reflecting where other types of signals cannot operate, either by drawbacks or where fluid is displaced.**

A mathematical model is presented in this paper described in state space as a basic acoustic sensor description. The objective is to adjust the parameters allowing the acoustic device to describe a signal in its trajectory, representing in geophysical manner the cavity form. Therefore, the control is performed on the response of the acoustic sensor model, adjusted with a parameter estimation process. The simulation results counts convergence between the reference and identified signals.

***Keywords* – Sliding modes, estimation, acoustic transducer, control.**

I. INTRODUCTION

Acoustic processes are of particular interest in *geophysical processes* because they describe different phenomena in nature, such as the Doppler effects with its relative velocity between two bodies [4]; stochastic entropy [10] with its pernicious effects when the signal exceeds the allowable thresholds [7]. The resonance commonly applies different methodologies describing the motion of an embryo in a womb or an egg, or the liquid levels in an oil well. Physically, the identification process is based on electronic sensors measuring and building the forms, allowing industrial automatic inspection where man has no access, or the eye cannot see, as the water depth sensing is performed by a sonar [6], [8], [9].

To meet the demands of science, multiple types of sensors such as photoelectric, magnetic, inductive, nuclear, acoustic, among other have been created. The acoustic sensor has a great potential still to be exploited in applications such as: content description in deep wells, watersheds, lakes, caverns, applied in places where solutions have a very rough shape container [6].

In general, the challenges of all the acoustic sensors are:

a) The signal type must be consistent with the environment, where the moves measure permits describing the contents,

b) All types of sensors are designed for a range defined without self-adjusting.

An acoustic signal can be reflected in areas where other signal types cannot operate, either by the fluid drawbacks displacement or because they would require hard-reflected signals. For example, a distance of at least *500m* in aqueous media, operating with power lines at different speeds. Large distances require high capacity signals: In this paper a control model acoustic sensor response with a parameter estimator as an adaptive technique, solves the problem considered in the first application point; such as, the reflection intensity offering a high capacity self-regulation descriptor distance sensing the distance range without, previously described. In some sense only a range is identified, hence the device needs self-regulation range distance with a limited level of uncertainty, adjusting the piezoelectric reference parameters system, which is firstly compared with a piezoelectric model tracking. In the event, where the results need the self-adjusting and controlling law, the parameters set is used within the model like an adjustable parameters description for long distances and rough cavities considering the initial conditions reference system [6], [8] and, [9].

To overcome current limiting, the control action regulating the acoustic signal level emitted by a piezoelectric device ensures that the reflected echo signal requires that the acoustic sensor intensity reading, analyzes the reflected signal [1]. The model and the control system were the basis for a particular container description, without presupposing the distance, adjusting the parameters through the measurement system and control action.

Therefore, the piezoelectric model actuator according to [1] is simplified into the form:

*Research supported by National Polytechnic Institute, ESIME Zacatenco.
Julio César Tovar Rodríguez is with the National Polytechnic Institute, Mechanical and Electrical School, IPN av., México City 07738, México. (corresponding author to provide phone: 57296000 ext. 54660; e-mail: jctovar77@ hotmail.com).
Floriberto Ortíz Rodríguez, is with the National Polytechnic Institute, Mechanical and Electrical School, , IPN av., México City 07738, México (e-mail: betoor@hotmail.com).
Carlos Román Mariaca Gaspar is with the National Polytechnic Institute, Mechanical and Electrical School, IPN av., México City 07738, México (e-mail: carlos_mariaca@hotmail.com).

$$\begin{aligned} \dot{X}_t &= AX_t + Bw_t \\ Y_t &= CX_t + Dw_t \end{aligned} \quad (1)$$

Where the matrix order correspond to the differential equation order with $A \in \mathbb{R}^{[n\times n]}_{[0,-\infty[}$, $B \in \mathbb{R}^{[n\times 1]}_{[0,1[}$, $C \in \mathbb{R}^{[n\times n]}_{[1,-1[}$, $D \in \mathbb{R}^{[n\times 1]}_{[0,1[}$, $w_t \in N\{\mu_{w_t}, \sigma^2_{w_t} < \infty\}$

Theorem 1. *Let the model considered in in state space have the form (1). In agreement to (1) and [3], the recursive form is:*

$$\dot{Y}_t = GY_t + HV_t \quad (2)$$

Where: G, H *are matrices bounded with* $G \in \mathbb{R}^{[n\times n]}_{[0,-\infty[}$ *and,* $H = f(A, C, B, D)$, $V_t \in N\{\mu_{V_t}, \sigma^2_{V_t} < \infty\}$ **Proof** *(See Appendix).*

II. CONTROL LAW

Theorem 2. *The control law system* $V_t{}^*$ *with respect to (2) has the form:*

$$V_t^* = H^+\left(E_t - GY_t\right) \quad (3)$$

With H^+ *the pseudo-inverse matrix* H, *the innovation process* E_t *considered in (2), and* G, *as a unknown matrix having the form* $G \in \mathbb{R}^{[n\times n]}_{[0,-\infty[}$. **Proof** *(See Appendix).*

The piezoelectric device has a main problem which corresponds to parameter distance description [1], [4], described as $\hat{G}_t$. This is solved estimating the gain with a lower uncertainty in almost all points that make up its surface.

III. PARAMETERS ESTIMATION

Applying the control law (3) into the model described in (2), converges to reference system (Y_t) only if the matrix G[1], [3] is known. Unfortunately, the reference system viewed as a Black Box scheme [3]; the matrix gain G is unknown, because it corresponds to the internal system description. Consequently, the estimation process is required describing the internal matrix gain through the time process [2], [3], [5].

Theorem 3. *Let the recursive model (2), with answer have the stochastic matrix estimation based on* [2], [3], [13]-[17]:

$$\hat{G}_t = P_t Q_t \quad (4)$$

With P_t, Q_t *covariance and variance matrices, with respect to (2).* **Proof** *(See Appendix).*

Theorem 4. *Let* $\hat{G}_t$ *be a stationary process, the recursive form in discrete manner is (5) and is viewed in figure 1, based on* [2], [3].

$$\hat{G}_t = \alpha_t \hat{G}_{t-1} + \beta_t \quad (5)$$

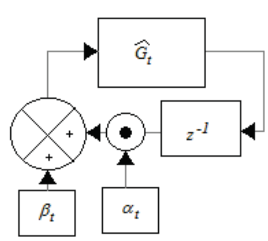


Figure 1 Recursive form of $\hat{G}_t$ considering stationary conditions.

A piezoelectric device developed as a mathematical model viewed as (1), considering the Black Box properties, only is observes input and output signals without knowing exactly the internally system operations. The control law does not affect the reference system, but the model through the parameters is estimated in the probability sense affecting the model converging to the reference piezoelectric device answer. All results are described in real numbers ($\mathbb{R}$), specifically over the hypothetical line that describes the container form.

The piezoelectric block diagram using the control action with adaptation into the model system is shown in figure 2.

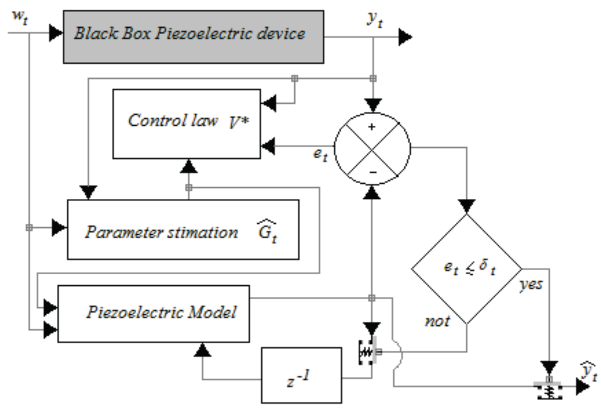


Figure 2 Piezoelectric system viewed as a control block diagram with parameters adjusted dynamically.

Now, the control law system V_t^* described in (3) considering (5) has the form:

$$V_t^* = H^+\left(E_t - \hat{G}_t Y_t\right) \tag{6}$$

With $H^+$the pseudo-inverse matrix H, the innovation process E_t considered in (2), and $\hat{G}_t$, as a matrix estimation.

IV. SIMULATION

The piezoelectric signal (y_t) with stochastic properties tracking the model output answer $(\hat{y}_t)$ with parameters estimation affecting the control law action. Both considered in (2) is illustrated in figure 3.

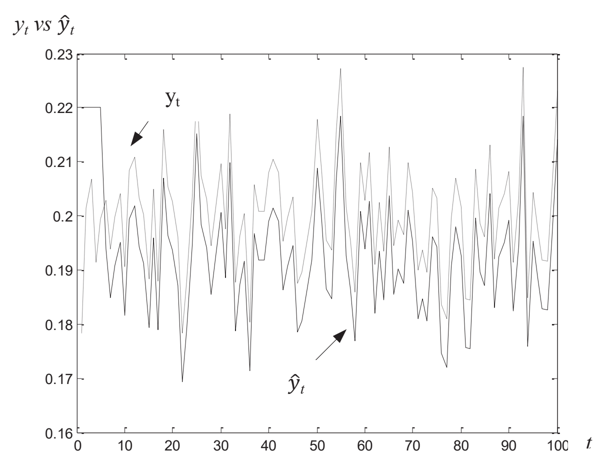


Figure 3 Piezoelectric temporal signal and its tracking.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

The adaptation scheme viewed in figure 1, was applied in the control law and into the model, observing that the answer is very narrow with respect to the real piezoelectric results. In figure 4, is observed the innovation process.

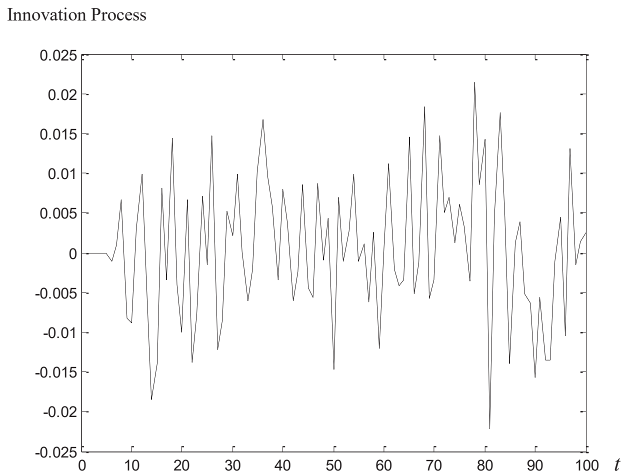


Figure 4 Innovation process through the time t

The convergence rate generated between the reference signal and the tracking output model is measured in decibels based on stochastic entropy and has the form $H_t = -20[e_t ln(e_t) - (1/20)H_{t-1}]$, with $e_t := y_t - \hat{y}_t$ and the result is viewed in figure 4.

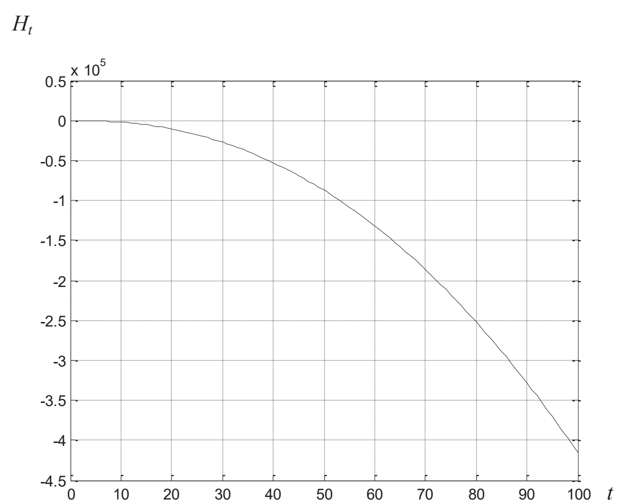


Figure 5. Stochastic entropy with respect to difference between the tracking and reference signal.

V. Conclusion

This paper presented a model considering into the references as a piezoelectric device description. The control system over the model required matrix gain parameters estimation, having an adaptive movement using into the tracking operations. The theoretical results were developed by the dynamical model properties and the control action. The simulation tracking was acceptable in probability sense, with convergence rate measured in decibels in stochastic description manner. The output piezoelectric device had an evolution with innovation signal. The tracking permitted a great convergence with stationary conditions affecting the control action over the model in positive form, minimizing the convergence error near to piezoelectric answer with rough conditions. The control law depended on the internal parameters, with adjustable gains estimation. The simulation results described the proposal answer, with a convergence level bounded in its movements as shown in the entropy.

Appendix

Proof (Theorem 1).
Let the model (1), with the first derivate described in:

$$\dot{Y}_t = C\dot{X}_t + D\dot{w}_t \tag{7}$$

Substituting in (7) to $\dot{X}_t$ de (1), has:

$$\dot{Y}_t = CAX_t + CBw + D\dot{w}_t \tag{8}$$

The internal state X_t in agreement to (1) is described in (9) with respect to observable signal:

$$X_t = C^{+}Y_t - C^{+}D\dot{w}_t \tag{9}$$

(9) in (8), has:

$$\dot{Y}_t = CAC^{+}Y_t - CAC^{+}Dw_t + CBw_t + D\dot{w}_t \tag{10}$$

In symbolic form (10) is described in (11) with $G \coloneqq CAC^{+}$, $H \coloneqq [(-CAC^{+} + CB) \quad D]$, and $V_t \coloneqq [w_t \quad \dot{w}_t]^T$.

$$\dot{Y}_t = GY_t + HV_t \tag{11}$$

Corresponding to (2).■

Proof (Theorem 2). Let the system (2) accomplish with:

$$M_t \dot{M}_t^T < 0 \tag{12}$$

Where the trajectory region with respect to the gain matrix M_t, is described in:

$$\dot{M}_t = -F_t \tag{13}$$

With F_t, a continuous function bounded by intervals with uniform measure t, accomplishing with the innovation process:

$$\hat{Y}_t = \dot{Y}_t + F_t \tag{14}$$

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Considering to (2) in (13) has:

$$-F_t = GY_t + HV_t - \dot{Y}_t \tag{15}$$

The difference between $\dot{\hat{Y}}_t$ and F_t in (15) is described in (16):

$$\left(F(t) - \dot{Y}(t)\right) = -GY_t - HV_t \tag{16}$$

The system innovation process (2) according to (16), it is described in:

$$E_t = Y_t + F_t \tag{17}$$

(17) in (16) and V_t, is developed in:

$$V_t^* = H^+\left(E_t^* - GY_t\right) \tag{18}$$

As a control law, described in (3). ■

Proof (Theorem 3).

Let $Z_t^T = f(Yt)$, such as applying the second probability moment (2) with respect to Z_t^T, has:

$$E\left\{\dot{Y}_t Z_t^T\right\} = GE\left\{Y_t Z_t^T\right\} + HE\left\{V_t Z_t^T\right\} \tag{19}$$

The mathematical operator properties in agreement to [2], [3] and, [5], the estimation is based on (19) has the form:

$$G_t = \left(E\left\{\dot{Y}_t Z_t^T\right\} - HE\left\{V_t Z_t^T\right\}\right) E\left\{Y_t Z_t^T\right\}^+ \tag{20}$$

Now, defining (20) with respect to (18) to P_t, Q_t in:

$$P_t := E\left\{\dot{Y}_t Z_t^T\right\} - HE\left\{V_t Z_t^T\right\}; Q_t := E\left\{Y_t Z_t^T\right\}^+ \tag{21}$$

And (21) in (20), the estimation matrix $\widehat{G}_t$ has the form:

$$G_t = P_t Q_t \tag{22}$$

Viewed in (4). ■

Proof (Theorem4). Let (5) has the components in agreement to (21), with stationary conditions P_t is described considering [2], [3], [5] in recursive form in differences:

$$P_t = \frac{1}{t^2}\sum_{i=1}^{t}\left(\dot{Y}_i Z_i^T\right) + \frac{H}{t^2}\sum_{i=1}^{t}\left(V_i Z_i^T\right) \tag{23}$$

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

With a delay, P_{t-1} is described in:

$$P_{t-1} = \frac{1}{(t-1)^2}\sum_{i=1}^{t-1}\left(\tilde{Y}_i Z_i^T\right) + \frac{H}{(t-1)^2}\sum_{i=1}^{t-1}\left(V_i Z_i^T\right) \tag{24}$$

(24) in (23) has:

$$P_t = \left(\frac{1}{t^2}\right)\left[\tilde{Y}_t Z_t^T + HV_t Z_t^T + (t-1)^2 P_{t-1}\right] \tag{25}$$

Considering that (22) delayed has the form:

$$P_{t-1} = G_{t-1} Q_{t-1}^{+} \tag{26}$$

And (23) in (21), has:

$$P_t = \left(\frac{1}{t^2}\right)\left[\tilde{Y}_t Z_t^T + HV_t Z_t^T + (t-1)^2 G_{t-1} Q_{t-1}^{+}\right] \tag{27}$$

(23) in (21), has:

$$G_t = \left(\frac{1}{t^2}\right)\left[\tilde{Y}_t Z_t^T + HV_t Z_t^T + (t-1)^2 G_{t-1} Q_{t-1}^{+}\right] Q_t \tag{28}$$

Minimizing (28) has:

$$G_t = \left(\frac{\tilde{Y}_t Z_t^T Q_t}{t^2}\right) + \left(\frac{HV_t Z_t^T Q_t}{t^2}\right) + \left(\frac{(t-1)^2 \tilde{\tilde{G}}_{t-1} Q_{t-1}^{+} Q_t}{t^2}\right) \tag{29}$$

(29) Symbolically is described in:

$$G_t = \alpha_t G_{t-1} + \beta_t \tag{30}$$

With α_t and β_t as:

$$\begin{aligned} \alpha_t &= \frac{(t-1)^2 G_{t-1} Q_{t-1}^{+} Q_t}{t^2} \\ \beta_t &= \left(\frac{\tilde{Y}_t Z_t^T Q_t}{t^2}\right) + \left(\frac{HV_t Z_t^T Q_t}{t^2}\right). \end{aligned} \tag{31}$$

And (30) is viewed in (5). ■

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

ACKNOWLEDGMENT

The authors would like to acknowledgment the financial support to ESIME Zacatenco, COFAA, EDI and CONACyT.

REFERENCES

[1] C. Pukdeboon, "Adaptive-Gain Second-Order Sliding Mode Control of Attitude Tracking of Flexible Spacecraft", Hindawi, Mathematical Problems in Engineering Vol. 2014, Article ID. 312494, 11 pages, 2014.

[2] J. J. Medel, R. Palma, "The diagonal Bernoulli differential estimation equation", RMF vol.59, no. 3, pp. 282-291, 2013.

[3] J. J. Medel, M. T. Zagaceta, "State space second order filter estimation", RMF, vol. 59, no.3, pp.254-263, 2013.

[4] J. Olaf P. P., "Relative-Velocity Distribution for Two Effusive Atomic Beams in Counter-propagating and Crossed-Beam Geometries", Advances in Mathematical Physics Vol. 2012, Article ID 906414, 18 pages.

[5] R. Palma, J. J. Medel y G. Garrido, "A m-dimensional Stochastic Estimator", RMF,vol. 58, no.1, pp. 69-71, 2012.

[6] J. Li, L. Yang, "Finite-Time Terminal Sliding Mode Tracking control for Piezoelectric Actuators", Abstract and Applied Analysis Vol. 2014, Article ID 760937, 9 pages.

[7] J. Chen et. al, "Link Prediction and Route Selection Based on Channel State Detection in UASNs", International Journal of Distributed Sensor Networks vol. 2011, Article ID. 939864,11 pages.

[8] S. J. Perez, J. A. Montero Díaz, S. Alcantara Iniesta, P. R. Hernandez, R. Castaneda G., "Optical sensing technique for Young's modulus measurements in piezoelectric materials" RMF, vol. 54, no. 3,pp 253-257, 2007.

[9] S. J. Perez, S. Alcantara I, P. R. Hernandez, R. Castaneda Guzman, "Sound speed resolved by photo-acoustic technique" RMF. Vol. 53, no. 3, pp. 213-217, 2007.

[10] L. Kuang et. al, "A numerical method for Analyzing Electromagnetic Scattering Properties of a Moving Conducting Object", International Journal of Antennas and Propagation, vol. 2014, article ID 386315, 9 pages.

[11] V. S. Pugachev., Probability theory and mathematical statistics for Engineers, Pergamon Press, Oxford, England, 1984.

[12] Haichen Qin et al, "An asymmetric Hysteresis Model and Parameter Identification Method for Piezoelectric Actuator", Mathematical problems in Engineering, vol. 2014, Article ID 932974, 14 pages

[13] F. Kwasniok, "Estimation of noise parameters in dynamical system identification with Kalman filters", Phys. Rev. E, vol. 86, no.036214, pp 1-8. 2012.

[14] X. Liu, Q. Gao, "Parameter estimation and control for a neural mass model based on the unscented Kalman filter", Phys. Rev. E vol. 88, no. 042905 pp, 1-10. 2013.

[15] F. Kwasniok, "Analysis and modeling of glacial climate transitions using simple dynamical systems", Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, vol. 371, no. 20110472, pp. 1-11, 2013.

[16] F. Kwasniok, "Estimation of noise parameters in dynamical system identification with Kalman filters", Phys. Rev. E, vol. 86, no. 036214, pp. 1-10, 2012.

[17] X. Liu, Q. Gao, "Parameter estimation and control for a neural mass model based on the unscented Kalman filter", Phys. Rev. E, vol. 88, no. 042905, pp. 1-8 2013.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 2, páginas 9-17.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Regularized divide and conquer training for dendrite morphological neurons*

Erik Zamora and Humberto Sossa

***Abstract*—We introduce regularization methods for training dendrite morphological neurons to avoid overfitting. Overfitting occurs if a learning model describes irrelevant noise instead of the fundamental relationship. We prevent this problem by limiting the depth of recursions in the proposed regularized training which is based on a divide and conquer strategy. We present the training algorithm in a recursive way as an efficient implementation. Experiments were performed based on several real and synthetic datasets. Results show that we can enhance the testing error and reduce the model complexity in classification tasks at low dimensionality.**

I. Introduction

Dendrite morphological neurons (DMN) are another way to model pattern classes. Classical perceptrons with only one layer divide the input space into two regions, using a hyperplane as a decision boundary. In the case more layers are presented, perceptrons divide the input space using a hyper-surface. In contrast, dendrite morphological neurons divide the same space by several hyperboxes that together can create complex nonlinear decisions boundaries allowing separating classes with only one neuron (for example in Figure 1). This is not possible with one-layer-perceptron. Other advantages appear during training DMNs: 1) there are no convergence problems; 2) training is performed in only one epoch; and 3) it is deterministic, that is, every time it is executed, we obtain the same result. All of these advantages are that result of using morphological operations (min and max) instead of addition and multiplication.

Recently, a training method for DMN based on divide and conquer strategy has been proposed in [1]. This method has shown better classification performance than other machine learning classical approaches, such as MLP with one hidden layer [2], support vector machines [3] and radial basis function networks [4]. However, the method described in [1] suffers from overfitting, because this approach is greedy, trying to classify perfectly every training pattern. In machine learning, overfitting occurs if a learning model describes irrelevant noise instead of finding data fundamental structure. An overfitted model has too many parameters relative to the number of training patterns and shows poor predictive performance. We illustrate this in Figure 1. Even when the non-regularized model has more hyperboxes than regularized model, this last simpler model presents a less testing error. In short, the non-regularized model tends to learn irrelevant variations in data affecting the predictive performance and learning generalization. In this paper, we propose using regularization methods to overcome this drawback of DMNs.

The rest of the paper is organized as follows. In section II, we provide a short review of the related work. In section III, we give a concise description of DMNs. In section IV, we introduce our proposed regularized training method. In section V, we present the experiments, results, as well as a short discussion to show the effectiveness of our proposal. Finally, in section VI, we give our conclusions and directions for future research.

*Research supported by the UPIITA and the CIC, both schools belonging to the Instituto Politécnico Nacional (IPN). This work was economically supported by SIP-IPN [grants numbers 20160945 and 20161116]; and CONACYT [grant numbers 155014 (Basic Research) and 65 (Frontiers of Science)].

Erik Zamora is with the Instituto Politécnico Nacional, UPIITA, Av. Instituto Politécnico Nacional 2580, Col. Barrio la Laguna Ticoman, Ciudad de México, 07340 (corresponding author to provide phone: 55-4588-7175; e-mail: ezamorag@ipn.mx).

Humberto Sossa is with the Instituto Politécnico Nacional, CIC, Av. Juan de Dios Batiz S/N, Col. Nueva Industrial Vallejo, Ciudad de México, 07738 (e-mail: hsossa@cic.ipn.mx).

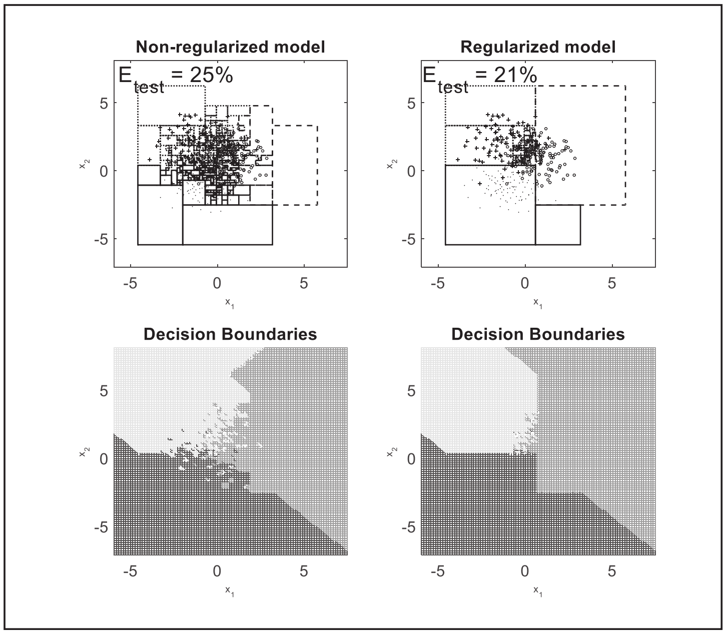


Figure 1: Overfitting problem for dendrite morphological neurons. Regularized model is less complex and has less testing error than non-regularized model.

II. Previous Work

Morphological neurons were proposed by Davidson and Hummer in their seminal paper [5]; their computing capabilities were studied by Ritter and Sussner [6]. Taking into account that information processing occurs also in dendrites, not only in cell body of neurons [7], the dendrite morphological neurons were proposed as an extension in [8]. A key issue is their training; we need to determine automatically the number of dendrites and the dendrite weight values. Several training approaches have been proposed, refer for example to [8, 9, 10, 11, 12, 1]. The training method based on divide and conquer strategy [1] has shown to outperformed Ritter's training approach [8] and other classical learning models. However, as we have mentioned in section I, this training approach suffers from overfitting as we will show in section V. In this paper, we enhance this method to overcome overfitting; we also present a recursive way for its efficient implementation. For more applications of DMNs, the reader is referred to [13].

III. Dendrite Morphological Neurons

A DMN has shown to be useful for pattern classification. It segments the input space into hyperboxes of N dimensions. In this work, we consider that each dendrite generates only one hyperbox assigned to a particular class; one important feature is that there is no overlap among hyperboxes. The output y of a neuron (see Figure 2) is a scalar given by

$$y = \text{argmax}_k(d_{n,k}), \qquad (1)$$

where n is the dendrite number, k is the class number, and $d_{n,k}$ is the scalar output of a dendrite given by

$$d_{n,k} = \min_i(\min(x-w^n_{min}, w^n_{max} - x)), \qquad (2)$$

where x is the input vector, w_{min} and w_{max} are dendrite weight vectors. The min operations together check if x is inside the hyperbox limited by w_{min} and w_{max} as the extreme points, refer to Figure 2. If $d_{n,k} > 0$, x is inside the hyperbox; if $d_{n,k} = 0$, x is somewhere in the hyperbox boundary; otherwise, it is outside. Note that the inner min operation in (2) compares elements of the same dimension between the two vectors, generating another vector, while the outer $\min_i$ operation takes the minimum among all dimensions, generating a scalar. A good property of DMNs is that they can create complex nonlinear decisions boundaries that allow separating classes with only one neuron (see Figure 1), in contrast to classical perceptrons. To be useful, perceptrons require to be arranged into layers that increase the artificial neural complexity and DMNs require more dendrites. It is worth mentioning that if (1) produces several maximums, the argmax operation takes the first maximum as index class to which the input pattern is assigned. Qualitatively speaking, this occurs when is equidistant to more than one hyperbox.

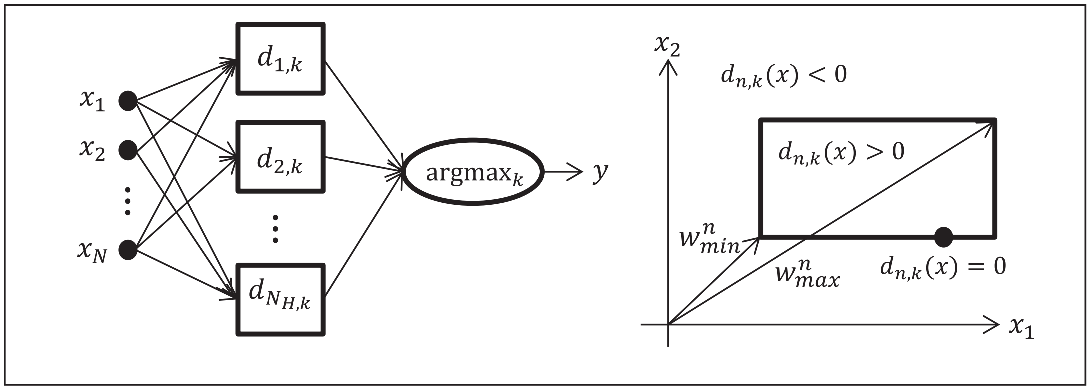


Figure 2: Dendrite morphological neuron and an example of a hyperbox in 2D generated by its dendrite weights. The hyperbox divides the input space for classification purposes.

IV. Divide and Conquer Training with Regularization

In this section, we extend the method in [1] by introducing regularization as a way to reduce overfitting obtaining a better performance as we will see in section V. Besides, we describe how the training method is implemented in a recursive way. The code and a demo of this training algorithm are online at [14].

Let as consider that we want to classify some patterns into N_c classes. We have a training samples set represented as a matrix $P \in \mathbf{R}^{N\text{x}Qtrain}$, where N is the number of features and Q_{train} is the number of training samples. The target class for each sample is contained in vector $T \in \mathbf{N}^{1xQtrain}$ where the membership class is given by a natural number. So we need a set of hyperboxes $List_H$ that enclose the training samples with their corresponding class $List_C$ as illustrated in Figure 3-(b). A hyperbox H is represented by its dendrite weights $H=(w_{min},w_{max})$. The training goal is to determine the number of hyperboxes and their weights that allow classifying an input pattern.

Training consists on only two procedures shown in lines (1) and (2) in Algorithm 1. As can be appreciated, the algorithm begins by opening an initial hyperbox H_0 that encloses all the samples with a margin distance M respect to the data extremes. Next the divide and conquer strategy is executed in recursive way as shown in lines (4) to (17) in Algorithm 1. Then the algorithm chooses a training sample p to generate a sub-hyperbox H_{sub} around it. Next it extracts the samples (P_{Hsub},T_{Hsub}) from (P,T) that are enclosed in H_{sub}. A new level L of recursion executes again with these extracted samples. An example of this recursion process is provided in Figure 3-(a).

The recursion divides H_0 until one of the following two cases happen: 1) the percentage error $E_{\%}$ in the hyperbox H is less or equal to an adjustable value E_0, or 2) the recursive level L plus one is greater than an adjustable value L_0. These adjustable values allow us to regularize the training, avoiding overfitting as well as generating complex models as we will describe in section V. The percentage error is defined as

$$E_{\%} = |P_{mode}|/|P| \qquad (3)$$

where P_{mode} is set of training samples in P with the class of statistical mode.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

At the end of the recursion process, the deepest hyperbox is assigned to the ruling class which is chosen by the statistical mode of T. The algorithm begins the recursive closing procedure by appending all generated sub-hyperboxes to $List_H$ with their corresponding class (see lines (13) and (14)) in Algorithm 1. Finally, the algorithm simplifies the number of hyperboxes by joining the hyperboxes sharing the dimensions.

Algorithm 1: This is the recursive algorithm that implements the regularized divide and conquer training for dendrite morphological neurons.

(1) $H_0 :=$ *generate_initial_hyperbox(P,M)*
(2) $(List_H, List_C) := D\&C(H_0, P, T)$
(3) ---
(4) $(List_H, List_C) := D\&C(H, P, T)$
(5) *if* $E_{\%} <= E_0$ *or* $L+1 > L_0$
(6) $List_H := H$
(7) $List_C :=$ *statistical_mode(T)*
(8) *else*
(9) *while P is not empty*
(10) $H_{sub} :=$ *generate_subhyperbox(H, p)*
(11) $(P_{Hsub}, T_{Hsub}, P, T) :=$ *extract_samples*(H_{sub}, P, T)
(12) $(List_{Hsub}, List_{Csub}) := D\&C(H_{sub}, P_{Hsub}, T_{Hsub})$
(13) $List_H := List_H \cup List_{Hsub}$
(14) $List_C := List_C \cup List_{Csub}$
(15) *end*
(16) $(List_H, List_C) :=$ *join_hyperboxes*$(List_H, List_C)$
(17) *end*

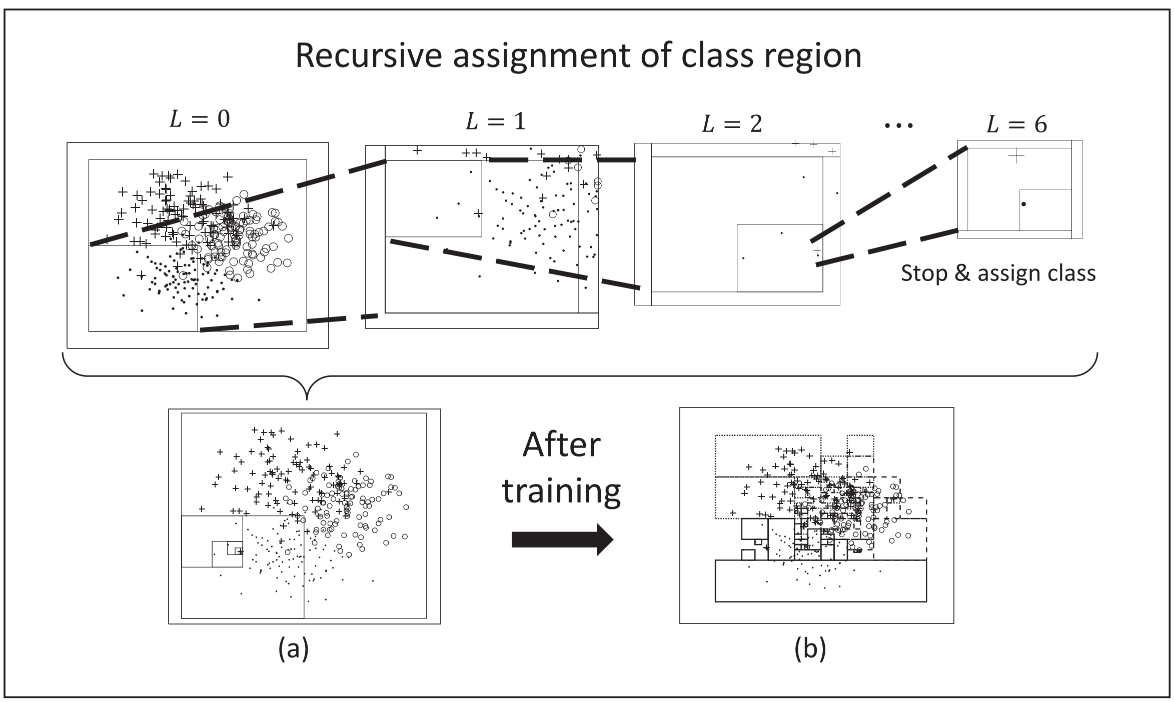


Figure 3: Divide and conquer training is explained.

V. Experiments and Results

In this section, we compare two regularization methods with different synthetic and real datasets. The results demonstrate the relevance of the proposed regularized training methods for DMNs. Furthermore, we discuss the advantages and disadvantages of each regularization method, trying to determine which the best is. The reader can reproduce the experimental results of this paper by using the online code provided at [14].

A. Synthetic and Real Datasets

Regularized training methods are first applied to synthetic datasets with high overlapping. We have chosen datasets where overlapping is presented because overfitting is more relevant in this kind of problems. The first synthetic dataset is called A, and it is generated by two Gaussian distributions with standard deviation equal to 0.9. The first class is centered around (0, 0); the second class is centered around (1, 1). The second synthetic dataset is called B and consist on three classes which follow a Gaussian distribution with standard deviation equal to 1.0. The classes centers are: (-1, -1), (1, 1), and (-1, 2). Figure 4 shows the distribution of these datasets and the overlapping among the classes.

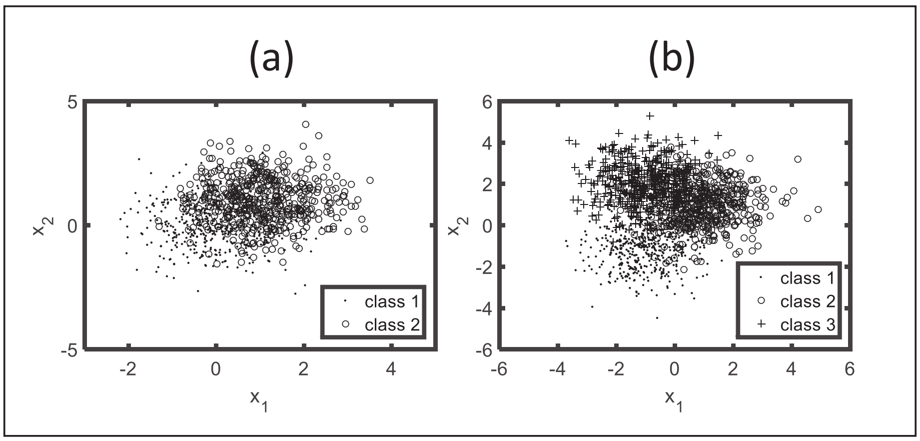


Figure 4: Plotting of the synthetic datasets showing overlapping between classes; (a) for A dataset, and (b) for B dataset.

We also use some real datasets to test the performance of the proposed regularized training methods. Most of these datasets were taken from the UCI Machine Learning Repository [15]. However, we also use MNIST [16] and CIFAR10 [17] datasets to have tasks with high dimensionality. We describe them briefly. The Iris dataset is a classical non-linearly separable problem with three types of iris plant and four describing features. The Liver dataset consists of detecting liver disorder (two classes) by means six features that relate blood tests and alcoholic drinking. The Glass dataset has the purpose to identify the glass type in crime scenes by means ten properties of glass. Ideally, the number of classes is seven, but no instances are provided for class number four in this dataset, refer to Figure 5). The Page Blocks dataset needs to classify the blocks of documents into five categories based on ten block geometrical features. The Letter Recognition dataset has 26 letters from different fonts which must be identified by 16 geometrical and statistical features. The Mice Protein expression dataset classifies the mice into eight classes based on features such as genotype, behavior and treatment. In this work, we only use the expression levels of 77 proteins as features for our experiments. Finally, we apply the proposed method to two image classification tasks taking the pixels as features (without intermediate feature extraction). The MNIST dataset is a classical recognition task for handwritten digit recognition (zero to nine) with image size of 28x28 pixels. The CIFAR10 requires recognizing ten among animals and vehicles by means of color images of 32x32 pixels. It is worth mentioning that in some of these datasets, we found duplicates; we deleted them for the experiments. Figure 5 shows the class distribution for each dataset.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

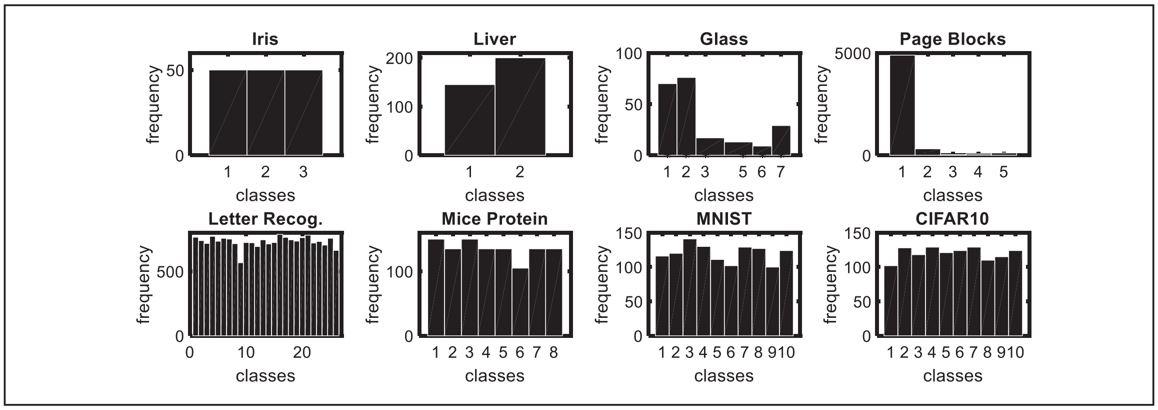


Figure 5: Class distribution for each real dataset. In the case of Glass and Page Blocks datasets, data distribution is clearly not uniform.

B. Regularization Curves

In this subsection, we analyze the performance of morphological neurons trained according to the tolerated error E_0 and the depth limit L_0 of the model. We measure performance in terms of the training error E_{train}, the testing error E_{test}, and the model complexity N_H/Q_{train}. This last quantity is defined as the ratio of the number of hyperboxes N_H by the number of training samples Q_{train}. If this ratio is equal to one, then the neuron uses a hyperbox for each training sample. In this extreme case, the model is very complex and presents no generalization. In contrast, if this ratio is close to zero, this means that the number of hyperboxes is much smaller than the number of training samples. Therefore, the model is simple and generalizes well. Ideally, we are looking for a morphological neuron with the smallest testing error E_{test} and with relatively few hyperboxes ($N_H/Q_{train} \approx 0$), that is, a simple and well generalized learning model.

We decided to vary one hyperparameter (L_0 or E_0) at the time to find out which one of regularization procedure is the most convenient: 1) regularization by the tolerated error; and 2) regularization by limiting the model depth. So regularization generates curves for two cases: 1) maintaining $L_0=\infty$, varying the value of E_0 from zero to one, and 2) maintaining $E_0=0$, we change the value of L_0 from 0 to 11, where $L_0=0$ means that the initial hyperbox is not divided and a single dendrite is used in neuron (11 because after that value no changes are manifested in these datasets). Figures 6 and 7 show the results obtained for each database, for each of these cases. Margin M is set to 0.1 for all experiments.

We have observed that the model complexity behavior is monotonic: complexity decreases as more tolerated error is allowed in the hyperboxes (Figure 6), and increases as more divisions are allowed in the hyperboxes (Figure 7). In contrast, the behavior of the training and testing errors is not monotonic. These errors increase and decrease depending on the specific dataset. For each dataset, in the first row of Figure 6 (except for the Iris case), the smallest testing error occurs somewhere when $E_0>0$, so the regularization improves the testing error; it also reduces model complexity. However, there is no value of E_0 better than zero that can improve the testing error for the datasets as shown in the second row of Figure 6 and for the Iris case. The regularization by E_0 is not useful in these last datasets. We observed equivalent behavior for regularization curves by L_0. In the first row in Figure 7 (except for the Iris case), the best value for L_0 is less than the maximum value of L_0 which is 11. This means that regularization improves the testing error and lowers model complexity. In the second row in Figure 7 (and for the Iris case), limiting the model depth does not improve the testing error.

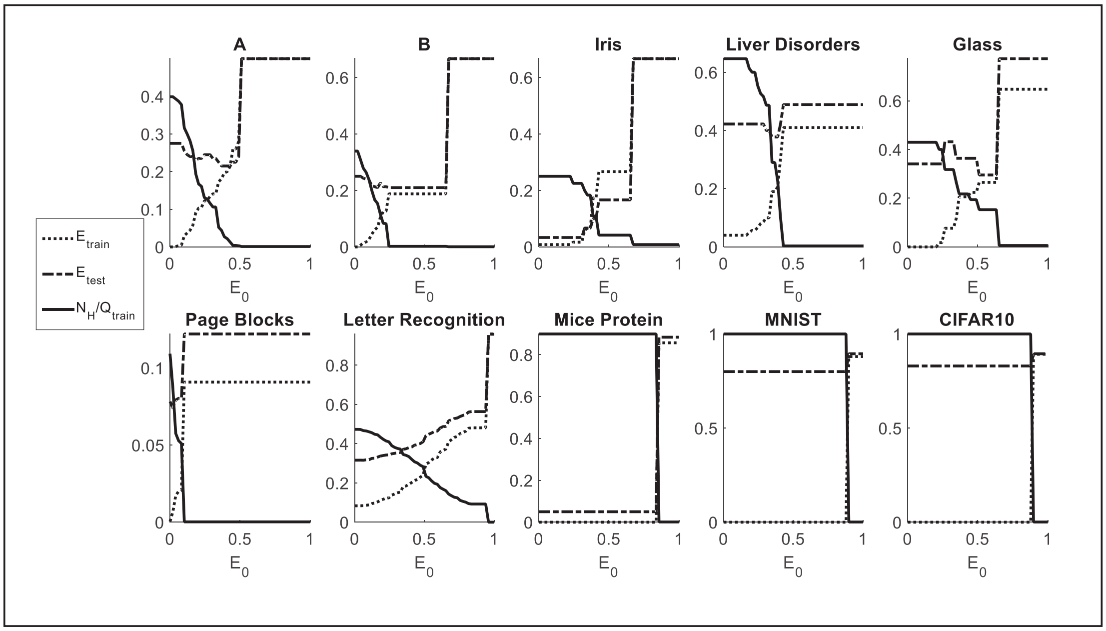


Figure 6: Regularization curves by E_0.

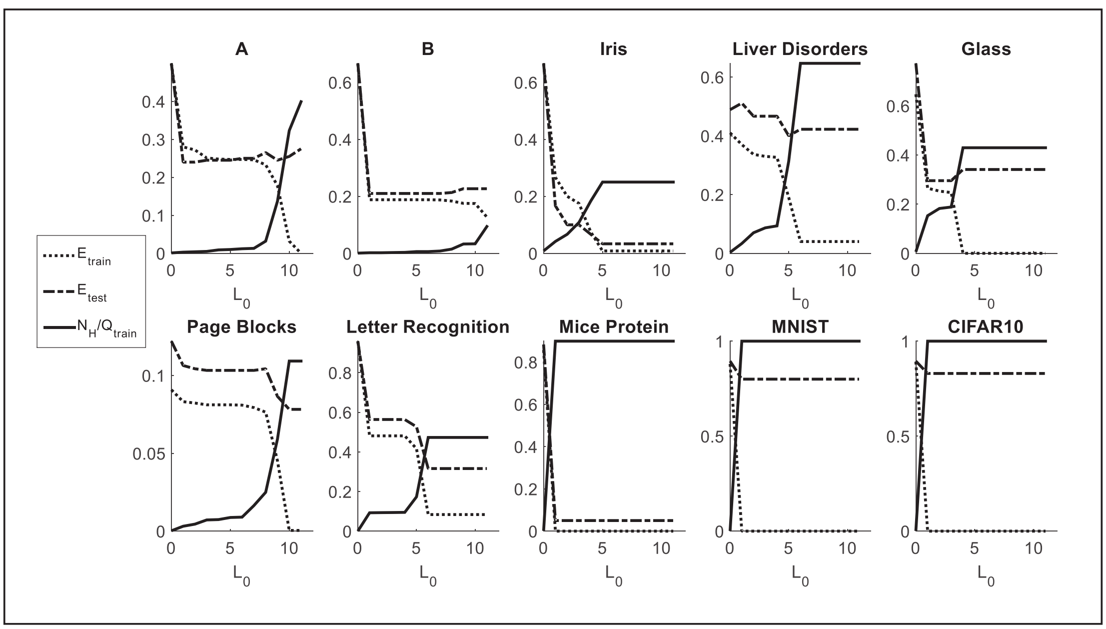


Figure 7: Regularization curves by L_0.

Table I compares the performance of the two proposed regularized training methods in three cases for each dataset: 1) we fix $E_0 = 0$ and $L_0 = \infty$; this case is equivalent to the original algorithm given in [1]; 2) we minimize the testing error by founding the best value of E_0 in the regularization curves in Figure 6; and 3) we do the same for L_0, by minimizing the testing error in regularization curves in Figure 7. For the case of datasets: A, B, Liver disorders and Glass, the testing error is improved approximately by 4%. For the case of datasets: Iris, Page Blocks and Letter Recognition, no improvement in testing error was

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

obtained. However, in the last three cases, model complexity is reduced by our proposal. These results show that the proposed regularization works and improves the performance of trained morphological neurons. Besides, if we compare the performance obtained by tolerating errors versus limiting model depth, we can find that the first method tends to be better than the second one. This is because the first method can be tuned more finely.

Nevertheless, we can see that regularization does not have any effect in the performance in the case of datasets with high dimensionality such as Mice Protein, MNIST and CIFAR10. This is due to the sparsity of the training samples in respect to the sub-hyperboxes generated by the initial hyperbox division. It is easy to see that the number of sub-hyperboxes increases exponentially 2^N as N increases. So, the number of sub-hyperboxes is much greater than the number of training samples at high dimensionality. Consequently, when we apply divide and conquer training, almost a hyperbox for each training sample will be generated. The morphological neurons for these three datasets have overfitting and their learning model is too complex. For each dataset, the first row shows the results obtained by the original algorithm [1]; the second row is obtained by the best value of E_0 and the last row is obtained by the best value of L_0. The best testing errors and complexities values are shown in bold.

Table I: For each dataset, the first row shows the results obtained by the original algorithm [1]; the second row is obtained by the best value of E_0; and the last row is obtained by the best value of L_0. The best testing errors and complexities values are shown in bold.

Dataset	N	Q_{train}	Q_{test}	E_0	L_0	N_H	L	E_{train} %	E_{test} %	N_H/Q_{train}
A	2	1000	200	0.0	∞	399	10	0.00	27.50	0.39
				0.42	∞	13	5	21.40	**21.50**	0.01
				0.0	2	4	2	27.20	24.00	**0.00**
B	2	1500	300	0.0	∞	509	11	0.00	25.00	0.33
				0.40	∞	3	1	18.80	**21.00**	**0.00**
				0.0	1	3	1	18.80	**21.00**	**0.00**
Iris	4	120	30	0.0	∞	30	4	0.80	3.30	0.25
				0.30	∞	27	4	1.67	**3.30**	**0.22**
				0.0	5	30	4	0.80	**3.30**	0.25
Liver Disorders	6	300	45	0.0	∞	194	5	4.00	42.20	0.64
				0.38	∞	72	4	19.70	**37.80**	**0.24**
				0.0	5	94	5	19.00	40.00	0.31
Glass	10	170	44	0.0	∞	73	3	0.00	34.10	0.42
				0.51	∞	26	1	26.50	**29.60**	**0.15**
				0.0	1	26	1	26.50	**29.60**	**0.15**
Page Blocks	10	4450	959	0.0	∞	486	9	0.00	**7.80**	0.10
				0.06	∞	235	9	2.00	8.00	**0.05**
				0.0	10	486	9	0.00	**7.80**	0.10
Letter Recognition	16	15071	3917	0.0	∞	7126	5	8.30	**31.50**	0.47
				0.06	∞	7032	5	8.45	31.55	**0.46**
				0.0	6	7126	5	8.36	31.58	0.47
Mice Protein Expression	77	900	180	0.0	∞	809	1	0.00	5.00	0.89
				0.8	∞	809	1	0.00	5.00	0.89
				0.0	1	809	1	0.00	5.00	0.89
MNIST	784	1000	200	0.0	∞	1000	1	0.00	80.00	1.00
				0.8	∞	1000	1	0.00	80.00	1.00
				0.0	2	1000	1	0.00	80.00	1.00
CIFAR10	3072	1000	200	0.0	∞	1000	1	0.00	83.00	1.00
				0.8	∞	1000	1	0.00	83.00	1.00
				0.0	2	1000	1	0.00	83.00	1.00

VI. Conclusion

In this paper, we presented two methods to regularize the training of dendrite morphological neurons. As we have seen, this allows reducing overfitting as well as the number of neuron dendrites. We observe that the best performance results were obtained by tolerating a maximum percentage error in each hyperbox. This can be useful for pattern classification at low dimensionality. However, when high dimensionality is presented none of the regularization methods reduces overfitting or lowers the number of hyperboxes. This remains an open issue for future investigation.

Acknowledgment

The authors would like to acknowledge the support for this research work by the UPIITA and the CIC, both schools belonging to the Instituto Politécnico Nacional (IPN). This work was economically supported by SIP-IPN [grants numbers 20160945 and 20161116]; and CONACYT [grant numbers 155014 (Basic Research) and 65 (Frontiers of Science)].

References

[1] H. Sossa and E. Guevara, "Efficient training for dendrite morphological neural networks," Neurocomputing, vol. 131, pp. 132 – 142, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231213010916

[2] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1,"D. E. Rumelhart, J. L. McClelland, and C. PDP Research Group, Eds. Cambridge, MA, USA: MIT Press, 1986, ch. Learning Internal Representations by Error Propagation, pp. 318–362.

[3] C. Cortes and V. Vapnik, "Support-vector networks," Machine Learning, vol. 20, no. 3, pp. 273–297, 1995.

[4] D. S. Broomhead and D. Lowe, "Multivariable Functional Interpolation and Adaptive Networks," Complex Systems 2, pp. 321–355, 1988.

[5] J. L. Davidson and F. Hummer, "Morphology neural networks: An introduction with applications," Circuits, Systems and Signal Processing, vol. 12, no. 2, pp. 177–210, 1993.

[6] G. X. Ritter and P. Sussner, "An introduction to morphological neural networks," in Pattern Recognition, 1996., Proceedings of the 13th International Conference on, vol. 4, Aug 1996, pp. 709–717 vol.4.

[7] I. Segev, "The handbook of brain theory and neural networks," M. A. Arbib, Ed. Cambridge, MA, USA: MIT Press, 1998, ch. Dendritic Processing, pp. 282–289.

[8] G. X. Ritter and G. Urcid, "Lattice algebra approach to single-neuron computation," IEEE Transactions on Neural Networks, vol. 14, no. 2, pp. 282–295, Mar 2003.

[9] R. Barron, H. Sossa, and H. Cortes, Morphological Neural Networks with Dendrite Computation: A Geometrical Approach. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 588–595.

[10] G. X. Ritter and G. Urcid, Computational Intelligence Based on Lattice Theory. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, ch. Learning in Lattice Neural Networks that Employ Dendritic Computing, pp. 25–44.

[11] D. Chyzhyk and M. Gra˜na, Optimal Hyperbox Shrinking in Dendritic Computing Applied to Alzheimer's Disease Detection in MRI. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 543–550.

[12] P. Sussner and E. L. Esmi, "Morphological perceptrons with competitive learning: Lattice-theoretical framework and constructive learning algorithm," Information Sciences, vol. 181, no. 10, pp. 1929 – 1950, 2011, special Issue on Information Engineering Applications Based on Lattices.

[13] H. Sossa and E. Guevara, Pattern Recognition: 5th Mexican Conference, MCPR 2013, Quer´etaro, Mexico, June 26-29, 2013. Proceedings. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, ch. Modified Dendrite Morphological Neural Network Applied to 3D Object Recognition, pp. 314–324.

[14] E. Zamora, "Regularized divide and conquer training for dendrite morphological neurons - Code," https://github.com/ezamorag/RegularizedDCtrainingforDMN, 2016.

[15] D. N. A. Asuncion, "UCI Machine Learning Repository. Irvine, CA: University of California, Department of Information and Computer Science," http://www.ics.uci.edu/ mlearn/MLRepository.html, 2007.

[16] C. B. Y. LeCun, C. Corinna, " The MNIST dataset of handwritten digits. NewYork: NYU, Google Labs and Microsoft Research," http://yann.lecun.com/exdb/mnist/, 1998.

[17] A. Krizhevsky, "Learning Multiple Layers of Features from Tiny Images. MIT and NYU," https://www.cs.toronto.edu/ kriz/cifar.html, 2009.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 3, páginas 18 - 32.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Wall-bot: the hexapod robot for inspection

E. C. Orozco Magdaleno, E. Castillo Castañeda, F. A. Aguirre Cerrillo, J. Franco Acuña, K. H. Floreán Aquino, E. Francisco Agustín

***Abstract*— This document presents the design, construction and control of the first version of WALL-BOT, a tele-operated hexapod robot for visual inspection in vertical surfaces. WALL-BOT is a bio-inspired prototype that consists in six open kinematics chains joined to a rigid chest and uniformly distributed, where the lead and rear legs are composed by four degrees of freedom (DOF) and the lateral legs by three DOF. The locomotion is designed for walking at vertical surfaces; therefore the hexapod has a suction system by suction cups. Since it is an inspection robot, WALL-BOT has a stereoscopic camera which is able to capture image and video in real time through a GUI (Graphical User Interface) developed in MATLAB. The robot is tele-operated from a gamepad and the programming algorithm is done in MATLAB.**

I. INTRODUCTION

In robotics of inspection the type of locomotion is implemented depending on the conditions of the ground where the robot will displace, nevertheless, different techniques have been implemented about bio-inspired locomotion owing to their high efficiency. There are developed prototypes for locomotion in vertical surface that depending on the design conditions, they have been done different strategies and configurations that solve specifics issues.

In relation to walking hexapod robots there are some works like <Behavior-based Modelling of Hexapod Locomotion> [1], that compares the walking insect and the six legs robots, explaining why they required a simultaneous control for eighteen actuators or more, taking in consideration the external conditions such as friction and the unpredictable slope.

With the purpose to orient and regulate de oscillating motion, the simplest scheme that has been proposed for legs movement modelling corresponds to: three angles of the joints that describe the actual position, and, three angles for destination that describe the position of the end-effector in the support surface.

Also, control algorithms have been implemented like that presented in <Complex-Order Dynamics in Hexapod Locomotion> [2], where the dynamics of interaction legs-ground in hexapod locomotion system is studied.

In normal conditions, the walking period could be difficult to keep it. In that case, is better to use a walking mode that can move any leg at any moment. These walking sequences are known as free sequences, like the alternating tripod. This walk can reach the theoretical maximum speed and the optimal walk for stability [3]. In the Fig. 1 is shown an instant of time where the sequence of alternating tripod becomes clear, sampling the projection of the mass center from the support polygon.

Research supported by IPN.
Ernesto Christian Orozco Magdaleno, student of Lazaro Cardenas Institute of Technology, Mexico (e-mail: chriztian_orz@hotmail.com).
Eduardo Castillo Castañeda, research professor of the Research Center in Applied Science and Advanced Technology from IPN, Mexico (e-mail: ecasilloca@ipn.mx).
Fabio Abel Aguirre Cerrillo, professor of Lazaro Cardenas Institute of Technology, Mexico (e-mail: fabio4aguirre@yahoo.com.mx).
Jonnathan Franco Acuña, student of the Research Center in Applied Science and Advanced Technology from IPN, Mexico.
Kevin Herubiel Floreán Aquino, student of the Mixteca´s University of Technology, Mexico.
Erick Francisco Agustin, student of the Mixteca´s University of Technology, México.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

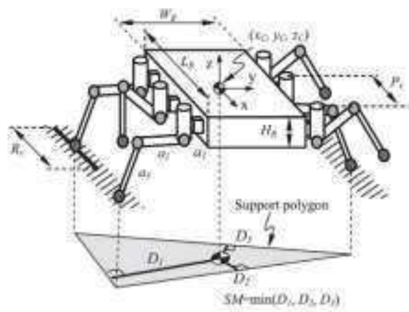


Figure 1. Alternating tripod and the support polygon [3].

A. Vertical Surface Locomotion

The bio-inspired climber robots that depend of adhesive systems for displacement, have become an essential tool for many tasks at industry, like the maintenance of the US navy aircraft carriers or the cleaning of high buildings [4], creating applications for the close future. In recent years, the research has mainly focused in the modelling of micro-scale adhesive phenomena and they have been developed experimental jobs with the final purpose to define the parameters of a specific model.

This type of robots that depends of dry adhesives requires lay on their legs against the wall for increase the surface contact and thus maximizes the adhesion force. Inappropriate force redistribution can cause that the robot falls down from the wall.

The robot´s optimal position is when it sticks with just five legs in a wall. The first leg is the red one, and gets in movement phase [5]. The optimal position become when the mass center is near to the leads legs, and the rears legs are extended, Fig. 2.

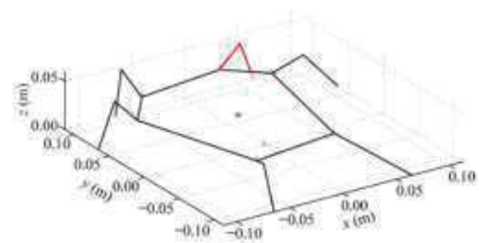


Figure 2. Optimal position with dry adhesives [5].

Another performed work for vertical surface locomotion is <A Six-Legged Walking Climbing Robot to Perform Inspection Tasks on Vertical Surfaces> [6], which presents the kinematic analysis of a climbing robot called “Hex-piderix”, which has six legs with three degrees of freedom each one, also in the end of each leg has a suction cup for walking in vertical surfaces. In Fig. 3 is shown the prototype and in Fig. 4 the sequence of the locomotion used to displace the robot.



Figure 3. Hex-piderix robot: Stability poligon [6].

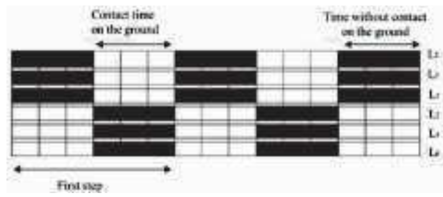


Figure 4. Hex-piderix robot: Tripod gait [6].

For this research is selected a suction system by suction cups and an alternating tripod locomotion sequence, offering to WALL-BOT a strong suction at the surface. The six legs robot walk has been design using a quasi-static model that takes into account the structure of the robot and the adhesive material features.

II. DESIGN AND CONSTRUCTION

The main feature of WALL-BOT is its bio-inspired mobility, this means that it morphology has characteristics as a living being. In this case, the cockroach is taken as reference because it has ability to walk in inclined surface and can travel in any direction, including the transition in orthogonal surfaces [7]. The legs configuration of the robot and the degrees of freedom of each one have been made by watching the movement of the legs when the cockroach is moving, this is seen in Fig. 5.



Figure 5. Legs configuration of a cockroach [7].

The configuration design consists in a six-chain open kinematics mechanism linked to a rigid chest having as characteristic that the lead legs and the rear legs <push-pull>, while the lateral legs help for the displacement and allow the alternating tripod locomotion sequence. The rear and lead legs are formed by four rotational joints (servomotors), where the first three are responsible to locate the suction cup at the space and the last joint guides it. The lateral legs are formed by three rotational joints, where the last one, as the other legs, guides the suction cup. Fig. 6 samples the schematic of the legs configuration, which are located in symmetrical form at the chest.

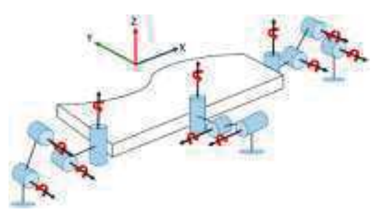


Figure 6. Schematic of legs configuration [7].

A. Suction System

The legs design is made for using a suction cup system. For this type of system is necessary to guarantee the parallel contact to the surface, as shown in Fig. 5. Thus, is necessary to guide the suction cup in any position of the path of locomotion, the responsible of this is a servomotor, which keeps the suction cup in parallel to surface at any time.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.



Figure 7. Suction cup in parallel to surface [7].

For the suction cup is necessary a pneumatic system providing compressed air to the legs of each tripod, so by this way the robot can walk in vertical surface. The pneumatic system is composed of six vacuum generators, six suction cups, two 3/2-way solenoid valves, and one air preparation unit, all the equipment is FESTO [7]. This system is divided in two parts, one part per tripod; see the distribution of the system in Fig. 8.

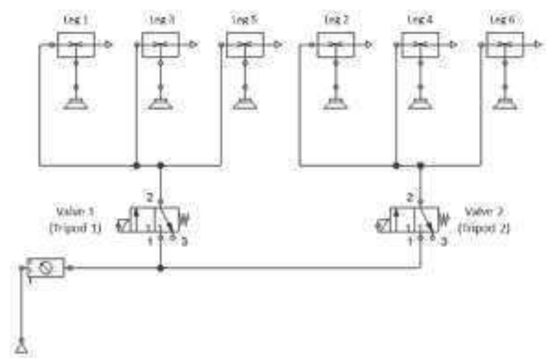


Figure 8. Pneumatic diagram of WALL-BOT´s subjection system [8].

B. CAD Modeling

The design of the prototype was performed using SolidWorks, where the parts of the robot are modeled and then assembled. In this software is analyzed the possible collisions between the joints and links, and also is define the dimensions of each part of the robot for manufacturing. The alternating tripod locomotion sequence was simulated; Fig. 9 shows the two phases of the sequence.

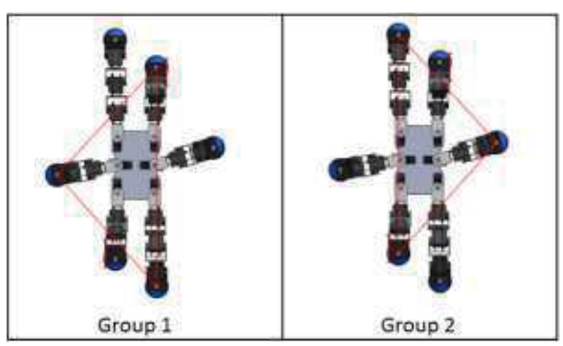


Figure 9. Groups of legs for the alternating tripod sequence [7].

C. Electronics

The controller is a 24-Channel Pololu Mini Maestro Servo Controller, which is specially designed to control servomotors. It has a USB interface and an internal command sequence control. The time resolution is 0.25 μs with a velocity and acceleration control function by frequency with a maximum range of 33 Hz in each channel.

The outputs of this module can be configured as PWM signals or as digital outputs. Thus, the digital outputs can send a signal to turn on the valves of the pneumatic system. To turn on the valves is implemented a circuit for each one with an optocoupler 4N35 and two transistors in Darlington configuration. A final transistor TIP31 supports a maximum current of one ampere. This circuit is shown in Fig. 10. The outputs of the controller are configured by MCC (Maestro Control Center). MCC is software of Pololu that allows controlling the servomotors connected in Mini Maestro, also MCC let configure the characteristics of the module.

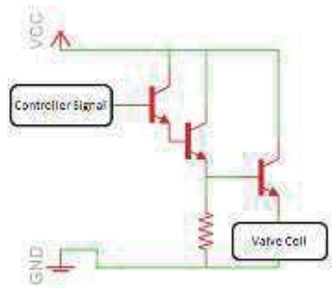


Figure 10. Optocoupler circuit diagram for one valve [7].

The actuators are TURNIGY TGY-901D servomotors. These servomotors work with PWM signals and have 13 kilograms per centimeter of torque with an operating voltage range of 6.0 to 7.2 volts, and speed of 6.54 to 7.48 rad/second. Also, they have metallic gears and ball-bearings.

D. Construction

A metallic box protects the controller and electronics; it is located at the front-central part of the chest and has a non-conducting cover to avoid damages in the electronic circuits. As inspection tool for the hexapod robot is mounted a stereoscopic camera over the box. Fig. 11 shows the hexapod robot.

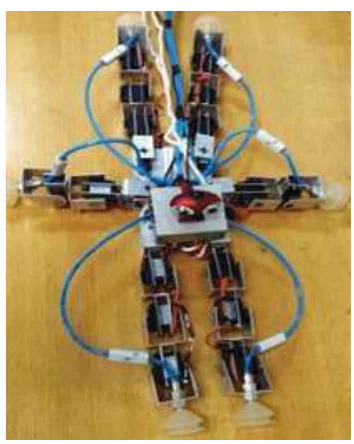


Figure 11. Hexapod robot.

III. INVERSE KINEMATIC MODELLING

For the robot´s motion is necessary to analyze the inverse kinematics of each joint and define the position equations. The representation of the hexapod robot legs and their reference system is shown in Fig. 12, it can be seen that the elements that hold the suction cup have a triangular form. The real measures in millimeters of each link are shown at Table I.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

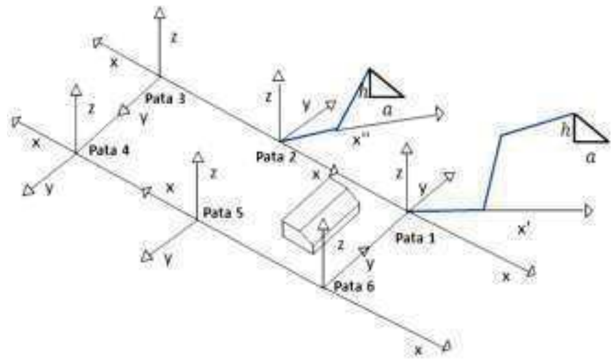


Figure 12. Representation the joints of the hexapod robot and their reference system [9].

TABLE I. MEASURES OF THE LINKS IN EACH JOINT [9].

Leg	L1 (mm)	L2 (mm)	L3 (mm)	a (mm)	h (mm)
1	781	780	785	275	660
2	765	520	N/A	300	680
3	791	773	772	297	661
4	779	780	780	310	660
5	780	527	N/A	300	680
6	780	772	773	311	660

For the kinematic analysis is considered that each joint is independent. At Fig. 13 is shown the coordinate axis system for the front and rear legs (1, 3, 4 and 6), as well as the angles formed by the links.

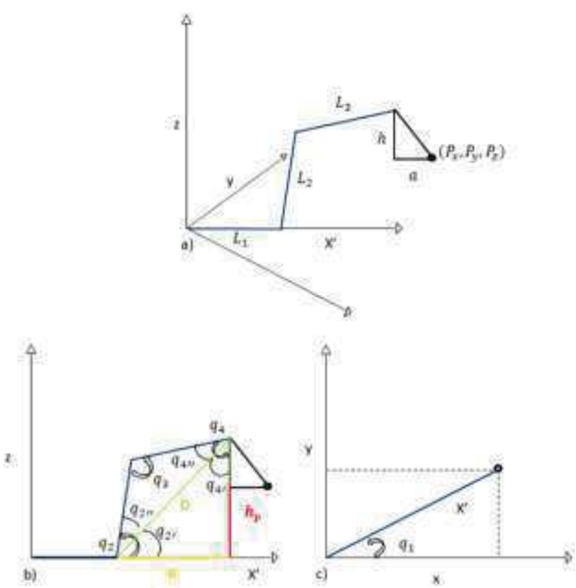


Figure 13. Coordinate axis system for front and rear legs: a) XYZ plane view , b) XZ plane view, c) XY plane view [9].

By geometrical considerations, equations (1) to (13) are deduced, which determine the value of actuator's angles.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

$$q_1 = tan^{-1}\left(\frac{P_y}{P_x}\right) \tag{1}$$

$$R = \sqrt{{P_x}^2 + {P_y}^2} - L_1 - a \tag{2}$$

$$h_p = P_z \tag{3}$$

$$D = \sqrt{R^2 + {h_p}^2} \tag{4}$$

$$q'_2 = tan^{-1}\left(\frac{h_p}{R}\right) \tag{5}$$

$$q''_2 = cos^{-1}\left(\frac{D^2 + {L_2}^2 - {L_3}^2}{2DL_2}\right) \tag{6}$$

$$q_2 = q'_2 + q''_2 \tag{7}$$

$$q_3 = 180 - 2q''_2 \tag{8}$$

$$q''_4 = q''_2 \tag{9}$$

$$q''_4 = cos^{-1}\left(\frac{D^2 + {L_3}^2 - {L_2}^2}{2DL_3}\right) \tag{10}$$

$$q''_4 = cos^{-1}\left(\frac{D^2 + {h_p}^2 - R^2}{2h_pD}\right) \tag{11}$$

$$q'_4 = 90 - q'_2 \tag{12}$$

$$q_4 = q'_4 + q''_4 \tag{13}$$

For lateral legs (1 y 2), the coordinate axis system and the angles formed by the links are shown in Fig. 14.

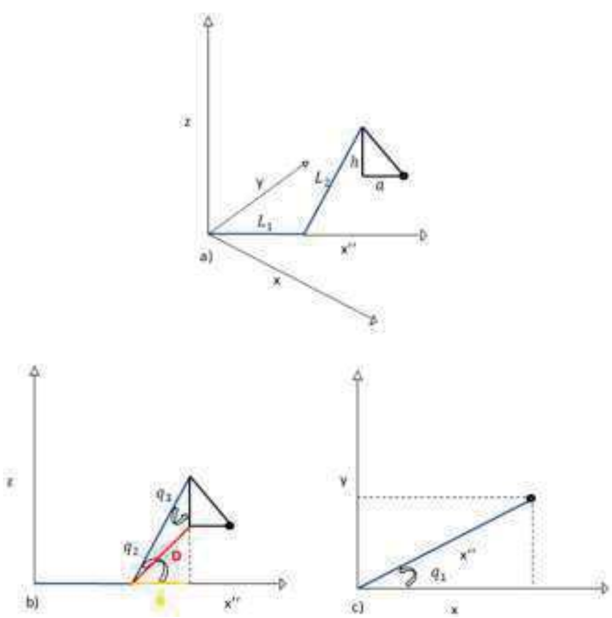


Figure 14. Coordinate axis system for lateral legs: a) XYZ plane view , b) XZ plane view, c) XY plane view [9].

The value of the angles is determinate with equations (14) to (18), which are obtained by geometric considerations.

$$q_1 = tan^{-1}\left(\frac{P_y}{P_x}\right) \tag{14}$$

$$R = \sqrt{{P_x}^2 + {P_y}^2} - L_1 - a \tag{15}$$

$$D = \sqrt{R^2 + {P_z}^2} \tag{16}$$

$$q_2 = tan^{-1}\left(\frac{P_z + h}{R}\right) \tag{17}$$

$$q_3 = cos^{-1}\left(\frac{{L_2}^2 + h^2 - D^2}{2L_2 h}\right) \tag{18}$$

IV. PATH PLANNING

The paths that the joints will follow are planned for a forward locomotion in straight line by the kinematic modelling developed.

A. Lateral Legs

The paths in Fig. 15 are selected for lateral legs, where the suction cup follows a rectangular path viewed from XZ plane and a circular path viewed from the XY plane. The arc formed by this path is too small that behaves as a straight line.

The path can be divided in two parts: the first one (blue path) is used to hold the suction cup in the surface and achieve the robot to displace along it, and the second one (red path) to take off the suction cup and locate it again in the initial position to restart the cycle, as is seen in Fig. 15. This path is the same for leg 1 and 2.

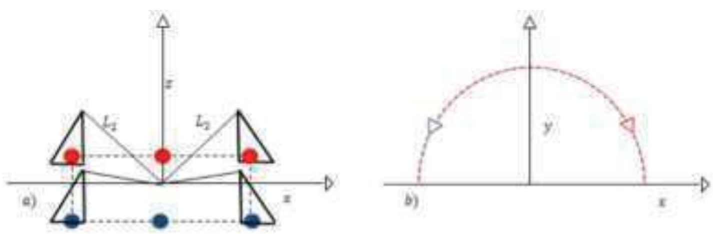


Figure 15. Lateral legs path: a) Rectangular in XZ plane , b) Circular in XY plane [9].

B. Lead and Rear Legs

The selected path for front locomotion is in pentagonal form, this viewed from XZ plane, Fig 16 a), and straight line viewed from XY plane as in Fig 16 b). This path allows to do a straight line walk, while the pentagonal form allows to sync up the properly way to sticking and unsticking the suction cup in each step.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

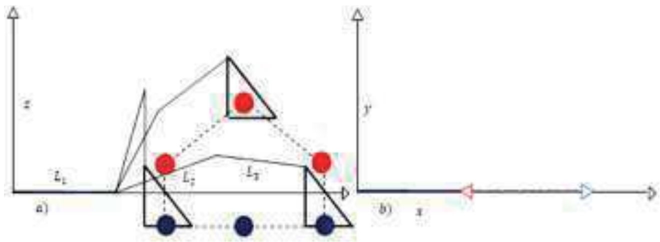


Figure 16. Rear legs path: a) Pentagonal in XZ plane, b) Straight line in XY plane [9].

In Fig. 16 is shown the path for the leg 3 and 4 (rear legs). As is shown, the upper part of the pentagon (red points) describes the path that joints follows to locate the suction cup, while the blue pointed path describes the path of the legs that put the suction cup in contact with the surface and the displacement along it once the suction cup is held.

For lead legs (1 and 6), the path is shown in Fig. 17. It can be seen that the upper part of the pentagon (blue pints) describes the path that joints follows to locate the suction cup, while the red pointed path describes the path of the legs that put the suction cup in contact with the surface and the displacement along it once the suction cup is held.

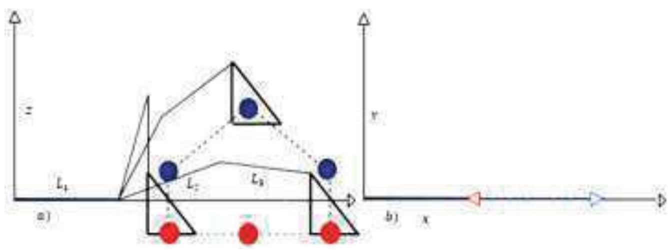


Figure 17. Lead legs path: a) Pentagonal in XZ plane, b) Straght line in XY plane [9].

V. PROGRAMMING AND CONTROL OF LOCOMOTION

The programming of the locomotion algorithm is done in MATLAB and it is divided in a main program and two subprograms. The main program controls the locomotion sequence from a gamepad, generating a tele-operation of the hexapod robot. The first subprogram calculates the inverse kinematics for a path of two steps, obtaining the value of the angles for all the joints. The second subprogram turns the angles from the inverse kinematics in to PWM signals for each position of the servos, and makes a serial communication format.

A. Main Program

This program (P1.m file) allows to tele-operate the hexapod robot using a JOINET gamepad. This gamepad has twelve buttons, in Fig. 18 is sample it distribution. Starting the program the buttons of the gamepad are read from de USB port of the PC. At pressing the START button the robot takes an initial position for being located in the wall, see Fig. 19, then the program waits the left joystick to be moved forward to start the locomotion.

The locomotion is divided in two steps, which means that every time the joystick is moving forward the robot takes a step, and if it is maintained, the robot takes a continuous march until the joystick is no longer pressed. It is important to mention that the program ensures complete steps and cannot fall down when the joystick is no longer pressed. To finish the locomotion is necessary to press the SELECT button and then press the button 1 to turn off the suction pneumatic system to take the robot from the wall.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

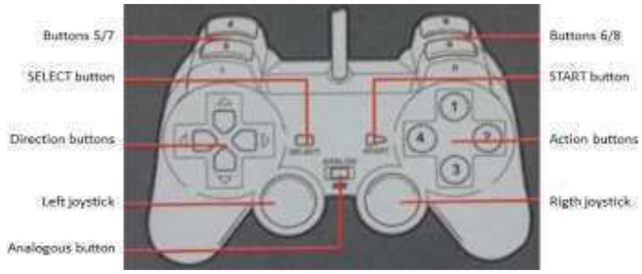


Figure 18. Buttons distribution of JOINET gamepad [8].



Figure 19. Robot´s initial position.

B. Inverse Kinematic Subprogram

A path cycle is equal to two steps, so the general path consists in the repetition of this pattern. The path cycle has 180 via points, thus one step has 90 via points. In accordance with the paths previously proposed, there are calculated 90 different angle values for each one of the 22 servomotors for each step (CI. m file).

There are calculated 90 different angles to take the robot from the position 1 to the position 2, and another 90 different angles to take the robot from the position 2 to the position 1, see Fig. 20. The result of this is a matrix of 180 values of different angles for each joint (matrix of 180X22). The angles are calculated with (1) to (18) of inverse kinematic.

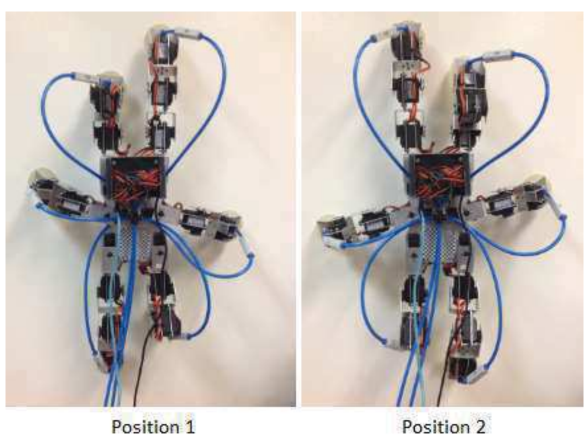


Figure 20. Main positions of the robot [9].

C. PWM Signals Conversion Subprogram

This subprogram (d2mcc.m file) obtains the pulse width in microseconds (μs) of the PWM signal that is necessary to take a servomotor to the angular position defined by inverse kinematic. Also, this subprogram

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

makes a format for serial communication with the controller. The conversion of the angular positions to PWM signals is done by (19):

$$PWM = \pm\frac{1000[\mu s]}{90°}(q) + b \tag{19}$$

Where q is the angular position to convert, PWM is the value of the pulse width in μs, b is the pulse width in μs corresponding to 0 within the angular operating range. The pending $\pm\frac{1000[\mu s]}{90°}$ is a common relation from the PWM signal which it is working (20 Hz).

The lineal relation for the conversion and the sign of the pending is shown in Fig. 21. This conversion is done for the 180 via points, a path cycle, of each one of the servomotors. The results are save in a matrix. The sending of these signals to the control card requires a serial communication protocol defined by the manufacturer.

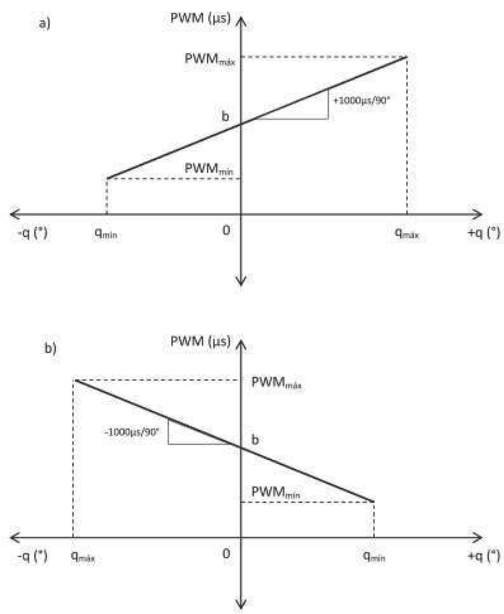


Figure 21. Conversion of angles to PWM signals: a) Positive pending, b) negative pending [9].

D. Communication Protocol

The Pololu MCC card generates PWM signals and sends it to the servo channels. The communication protocol is described below.

```
0xAA,num_disp,0x1F,num_canales,canal_inicial,PWM1_LB,PWM1_HB,PWM2_LB,PWM2_HB...
```

The first command byte 0xAA (170 in decimal) defines the band rate of the communication. Immediately is the number of the device, the device number for this controller is 12. The next command byte is 0x1F (31 in decimal) and defines the script of the position of each servomotor, then the number of channels to use, for this case are 22, the starting channel, 0 in this case, and finally the wanted position of each servomotor expressed in PWM signals. The signal value in μs is not usually sent in just one byte, so for this the signal value is multiplied by 4 and divided in low byte (LB) and high byte (HB). From the command signal of MATLAB is possible to send the string that defines the serial communication protocol by the next form:

```
fwrite(s2,[170,12,31,22,0,signal1_LB,signal1_HB]'uint8');
```

VI. IMAGE ACQUISITION GUI

To capture image and video it has been developed a GUI (Graphical User Interface) in MATLAB. This GUI takes image and video in real time and stores it in JPG and AVI format respectively. The acquisition is performed by the “Image Acquisition Toolbox” of MATLAB.

The video is capture at 40 fps (frames per second), this quantity was defined by testing the speed of the image acquisition, and 40 fps was the best. It is important to mention that although the cam is 3D, the image and video capture is performed in 2D format. The camera is a WEC-360 3D of SOMIKON with a maximum resolution of 640X480 per each lens, Fig. 22.



Figure 22. Main positions of the robot [10].

The GUI (Ad_Ima.m file) uses just one lens because both lens are practically at the same height and capture the same image. This GUI has the next characteristics:

- Capacity to capture images and videos in real time.
- Capacity to take continuous videos of 30 seconds each one, cleaning the flash memory of MATLAB at the end of each video. NOTE: This is because MATLAB does not support a long image acquisition.
- Capacity to take any number of images and videos.
- Capture mode selection (image or video), simple and fast.
- Simple interface, with indicators of number of videos and the recording time.
- Buttons for: start cam, finish cam, capture image, start recording video and finish recording video.
- Enable and disable of buttons to protect the GUI of a wrong use of them, for example, enable the recording buttons when the capture mode is on image, and vice versa.

In Fig. 23 is shown the Image Acquisition GUI and it can be seen the parts of it.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.



Figure 23. Image acquisition GUI [8].

VII. EXPERIMENTAL VALIDATION

As experimental validation of this research, performed a visual inspection was performed with the hexapod robot. To perform the inspection is necessary two computers and follow the next instructions:

1. Open the valve of compressed air.
2. Open MATLAB in both computers.
3. Connect the USB cable of the Mini Maestro card to the PC 1, and verify the COM number.
4. Connect the USB cable of the stereo camera to the PC 2, and verify that the resolution programmed match with the resolution available on MATLAB (the resolution can be different depending of the MATLAB version).
5. Connect the USB cable of the gamepad to the PC 1 and press the ANALOG button, wait for the red led to turns on.
6. Open MCC on PC 1.
7. Ensure that servos be disable in the STATUS window.
8. Open in PC1 P1.m, d2mcc.m and CI.m.
9. Open in PC 2 Ad_Ima.m.
10. Turn on the power supplies for the servomotors at 7 volts and for the pneumatic system ay 24 volts.
11. Run P1.m, to take the robot to the initial position is necessary to press the START button of the gamepad. NOTE: If is desired to cancel the operation press the SELECT button.
12. Place the robot on the wall.
13. Run Ad_Ima.m to start the GUI and start the cam.
14. Move the left joystick forward to start locomotion. If is desired to perform continues locomotion, just maintain forward the joystick.
15. To capture image or video, is just necessary to press the corresponding buttons of the GUI.
16. Press the SELECT button to finish locomotion.
17. Press button 1 to turn off the pneumatic system and can take down the robot.
18. Turn off the power supplies.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Following the previous sequence, vertical walking is performed with the hexapod robot along a wall of 2.5 meters of height (approximately) in the Laboratory of Mechatronics of the Research Center in Applied Science and Advanced Technology Unit Queretaro from the IPN. Fig. 24 shows some pictures of WALL-BOT during locomotion.



Figure 24. Vertical walking of WALL-BOT while is doing a visual inspection [8].

During the inspection were captured three videos. The videos and images captured with the GUI are stored in the folder of MATLAB where programming is running. At the Fig. 25 are shown the results from the image acquisition during the inspection.



Figure 25. GUI during inspection [8].

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

When the videos are in a playlist, for watching the inspection continuously, it can be appreciated a little lapse where the image lose the continuity (one second approximately). This lapse is because program has to store the video and clean the flash memory of MATLAB to start recording a new video.

VIII. CONCLUSION

This paper shows the design, construction, control and experimental validation of the first version of WALL-BOT. The developed project allows a viable and safe alternative for the people that works in dangerous high places, by this way they can do a visual inspection before climbing to perform tasks thus allowing to see if the work zone is safe or if is necessary a special equipment (suit for chemical spills, mask, etc.) and the necessary tools to work. All these tasks are simple and easy to do for the operator; it is just necessary two computers to run the programs.

The tele-operation through the gamepad is intuitive and does not require specialized training, so that any person can use this robot. With the above is given a big step in the handling of the robot, because in comparison with the automated systems the operator should has a special training to generate the paths that the robot has to follow, all this based in the inspection zone.

REFERENCES

[1] J. Schmitz and H. Cruse, “Behavior-based modelling of hexapod locomotion: linking biology and technical application”, *Arthropod Structure & Development*, vol. 33, pp. 237-250, April 2004.

[2] M. F. Silva *et al*, “Complex-order dynamics in hexapod locomotion”, *Signal Processing*, vol. 86, pp. 2785-2793, 2006.

[3] J. Estremera *et al*, “Continuous free-crab gaits for hexapod robots on a natural terrain with forbidden zones: An application to humanitarian demining”, *Robotics and Autonomous* Systems, vol. 58, pp. 700-711, 2010.

[4] V Robo Systems, “Suction-Adhering Self-Propelled Robots”. Available: http://www.urakami.co.jp/img/pdf/vrobo_brochure_en.pdf.

[5] P. Boscariol *et al*, “Optimal Gait for Bioinspired Climbing Robots Using Dry Adhesion: A Quasi-Static Investigation”, *Journal of Bionic Engineering*, vol. 10, pp. 1-11, 2013.

[6] X. Y. Sandoval-Castro *et al*, “Hex-piderix: A six-legged walking climbing robot to perform inspection tasks on vertical surfaces”, in *Proc. Of the 16th Int. Conf. CLAWAR-2013*, Sidney, pp. 399-407, 2013.

[7] J. Franco Acuña, “Diseño e implementación de un algoritmo para la locomoción en superficie vertical de un robot hexápodo”, M.T.A. thesis, Research Center in Applied Science and Advanced Tehcnology from IPN, Querétaro, Mexico, 2015.

[8] E. C. Orozco Magdaleno, “Validación experimental de un robot hexápodo para marcha vertical”, Lazaro Cardenas Institute of Technology, Lázaro Cárdenas, Mexico, 2015.

[9] K. H. Floreán Aquino and E. Francisco Agustín, “Diseño e implementación de un algoritmo de locomoción vertical para el robot WALL-BOT”, Mixteca´s University of Technology, Oaxaca, Mexico, 2015.

[10] Somikon (2016). *Somikon WEC-360.3D*. Available: http://www.somikon.de/3D-Webcam-PX-3602-919.shtml.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 4, páginas 33 - 40.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Effects of a reconfiguration on kinematic performance of a delta-type parallel robot

Albert L. Balmaceda-Santamaría, Eduardo Castillo-Castañeda

***Abstract*— This paper presents a new reconfiguration of a Delta-type parallel robot. The proposed reconfigurable mechanism, whose patent is in process, has a fixed platform comprising of a fixed element and mobile elements, which modify the size and shape thereof. Furthermore, the operational Cartesian workspace is increased thanks to the reconfiguration when it is com-pared with that of the original version. The input-output equation of the velocity of the reconfigurable mechanism is obtained by resorting to reciprocal-screw theory and the evident singular configurations are analyzed. Finally, the kinematic performance of the reconfigurable mechanism is presented, based on the calculation of a performance index. The results indicate that an improvement of the kinematic performance of the new robot is obtained, when this is compared with the performance of a Delta-type robot.**

I. Introduction

The flexible manufacturing system is a highly versatile production system, which can adapt to different requirements of the market. Reconfigurable manufacturing parallel manipulators can quickly adjust their structure according to a specific task, which is desirable to satisfy market needs and suitable for flexible manufacturing systems.

In robotics the reconfiguration concept was presented by [1] as the change of the characteristics of the robot in operation, distinguished in two types: static and dynamic. The main approaches that have been proposed for reconfiguration of parallel mechanisms as the Gough-Stewart platform are modular and variable geometry designs, see for instance [2-4].

A modular manipulator consists of a set of modules that can be assembled into robots with different attributes. The modularity has been used recursively in parallel manipulators by [5-8].

The reconfiguration of parallel manipulators using variable geometry consists of changing some dimensions of the robot with the purpose to generate new postures upon the same parallel manipulator. A double planar parallel reconfigurable manipulator presented by [9], the reconfigurable platform based on the Stewart-Gough of [10], a reconfigurable parallel mechanism with an adjustable base designed by [11] and a new family of reconfigurable parallel mechanisms [12], are some which may be mentioned.

Furthermore, many authors (including some aforementioned) have used the redundancy produced sometimes by reconfigurations of the parallel robots for improving characteristics such as: stiffness [13-15], force [16-17], accuracy [18], increased workspace and singularity-free workspace [4] y [19-22], as well as performance indices like manipulability, condition number, global condition, dexterity and global dexterity [23-26].

It is evident that reconfigurable parallel manipulators are a well-documented problem, since the reconfigurability remains today a challenge in the parallel robotic field, mainly in issues of occupy large structural spaces. In the case of the mechanisms aforementioned, they have a high level mechanical complexity in their structures and hence very expensive to be made.

In this paper a new reconfiguration of a 3-DOF Delta-type parallel robot, based on the concept of variable geometry is present-ed. The base robot used for the reconfiguration proposal, named Parallix LKF-2040, was designed as a didactic version at National Polytechnic Institute (IPN) in CICATA, Mexico [27]. The reconfiguration strategy is highly versatile, very simple and it is capable to return to the original configuration and mechanical characteristics of the Parallix LKF-2040.

The redundancy generated in the manipulator by the reconfiguration proposal, is used to improve the kinematic performance of the robot, by calculating a performance index along prescribed trajectories.

II. Description of the reconfigurable robot

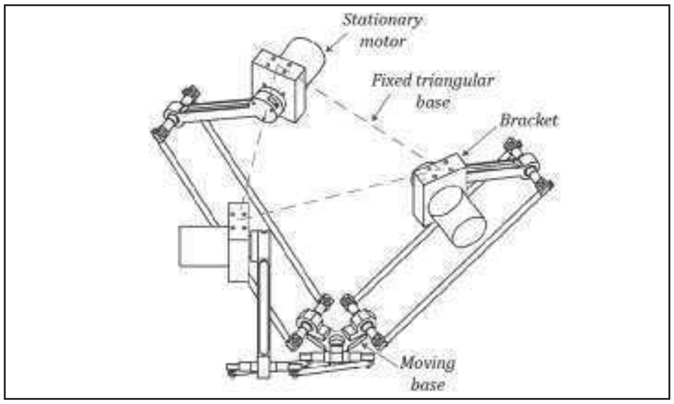
The Parallix LKF-2040 robot is a translational manipulator that was developed at IPN - CICATA. This manipulator is very used for teaching purposes, thanks to that it has an open architecture [27].

The Parallix robot comprises 3 stationary motors disposed angularly on the robot base through brackets. The motor axes are coupled to a kinematic chain at the bracket level. This mechanism has a structure of the well-known Delta robot, however the mechanism of Fig. 1 is a 3-RUU robot.

Albert L. Balmaceda-Santamaría, Universidad Nacional de Ingeniería, Managua, Nicaragua. (e-mail: albert.balmaceda@gmail.com).
Eduardo Castillo-Castañeda, Instituto Politécnico Nacional, Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada, Unidad Querétaro, Querétaro, México.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

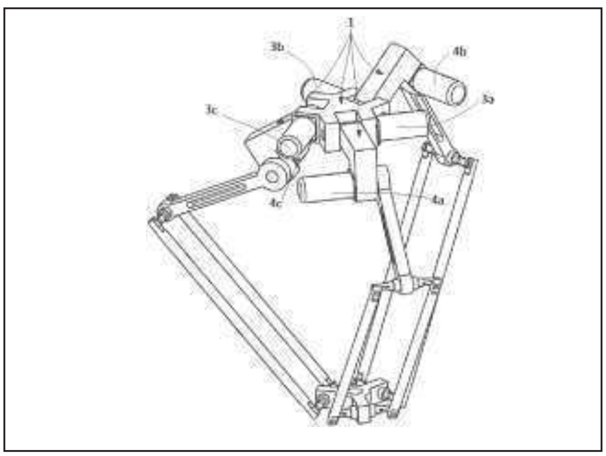
Figure 1. Current configuration of the robot Parallix



A. Reconfiguration proposal

The main variable that significantly influences in the Parallix workspace is the diameter of the fixed platform [28], since the diameter length directly impacts on workspace features, mechanical simplicity and number of actuators required for the reconfiguration. The following requirements were considered to implement the reconfiguration of Parallix robot: it must be mechanically simple, without adding weight and/or actuators into the kinematic chains, be able to change the entire workspace without modifying the lengths of links, increasing the versatility without to couple or uncouple links into mechanism and minimizing the structural space occupied by the robot. In Fig. 2, a conceptual design of reconfigurable Parallix LKF-2040 robot that fulfills the above conditions is shown.

Figure 2. Proposed reconfigurable mechanism

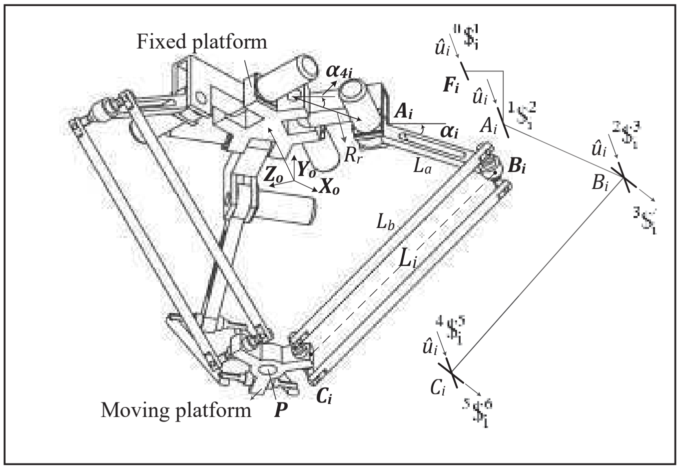


The reconfigurable manipulator of Fig. 2 comprises of a reconfigurable fixed platform (1) with a fixed element (2) and three mobile elements in form of framing square (2a, 2b, 2c) moved by the motors (3a, 3b, 3c). Such motors move the framing squares with simultaneous or independent movements between each other.

Each framing square modifies the radius and height between the axes of the motors (4a, 4b, 4c) and the center of the fixed element (2). It should be note that this reconfiguration is highly versatile, since the mechanism can be reconfigured also with each of the framing squares with decoupled movements from the rest of the mechanism and be able to return to the original configuration and mechanical characteristics of the Parallix LKF-2040 robot. In Fig. 3, the geometrical parameters of the reconfigurable manipulator Parallix are shown.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Figure 3. Geometrical parameters of the reconfigurable mechanism



III. Kinematic analysis

A. Direct displacement analysis

Direct displacement analysis consists of finding the location of the moving platform of coordinates $P = (Px\ Py\ Pz)$, given the generalized coordinates of the manipulator α_i. In what follows, the direct displacement analysis is performed based on the method presented by [29,33].

Now, let $X_oY_oZ_o$ be the fixed reference system in the center of the fixed platform of the manipulator. The coordinates of the points $F_i = (Fx_i\ Fy_i\ Fz_i)$, $A_i = (Ax_i\ Ay_i\ Az_i)$ y $B_i = (Bx_i\ By_i\ Bz_i)$ are calculated by projections as follows

$$F_i = [R_f \cos(\theta_i)\ \ 0\ \ R_f \sin(\theta_i)]$$

$$A_i = [R_r \cos(\alpha_{4i})\cos(\theta_i)+Fx_i\ \ R_r \sin(\alpha_{4i})\ \ R_r \cos(\alpha_{4i})\cos(\theta_i)+Fz_i]$$

$$B_i = [La \cos(\alpha_i)\cos(\theta_i)+Ax_i\ \ By_i\ \ La \cos(\alpha_i)\sin(\theta_i)+Az_i]$$

where i=1,2,3; R_f is the distance from the fixed reference system to each point F_i; θ_i is the orientation angle of each kinematic chain. Evidently $By_i=La \sin(\alpha_i)+ Ay_i$ is equal for all kinematic chain.

Now, consider that the moving platform is an equilateral triangle $C_1C_2C_3$ of side r and center P. Therefore, consider $C_1=(X\ Y\ Z)$, then coordinates C_2 and C_3 can be formulated according to C_1 as follows

$$C_{2,3}=(X+r \cos(\theta_{2,3})\ \ Y\ \ Z+r \sin(\theta_{2,3})) \quad (1)$$

therein $\theta_{2,3}$ is the orientation angle of kinematic chains 2 and 3, respectively.

In order to calculate the coordinates of X, Y and Z consider the following closure equations $(C_i - B_i)\bullet(C_i - B_i)=Lb^2$ that generate

$$X_2+Y_2+Z_2+K_{i1}X+K_{i2}Y+K_{i3}Z+K_{i4}=0 \quad (2)$$

In this way, $K_{ij}(j=1,2,3)$ are obtained according to the parameters and generalized coordinates of the mechanism, e.g., $K_{11}=-2Bx_1$, $K_{12}=-2By_1$, $K_{13}=-2Bz_1$, $K_{14}=Bz_1^2+By_1^2+Bx_1^2-Lb^2$…, etc. Furthermore following some computations on (2), the unknowns X and Y can be expressed in terms of Z as follows

$$X=W_1Z+W_2,\ \ Y=W_3Z+W_4 \quad (3)$$

where the coefficients $W1,\ldots,W4$ are calculated according to the coefficients K_{ij}, e.g.,

$$W_1 = \frac{K_{33}(K_{22}-K_{12})+K_{23}(K_{12}-K_{32})+K_{13}(K_{32}-K_{22})}{K_{21}(K_{32}-K_{12})+K_{11}(K_{32}-K_{22})+K_{31}(K_{22}-K_{12})} \quad (4)$$

Finally, substituting (3) in any of (2) produces a second-order equation in the unknown Z as follows

$$aZ^2+bZ+c=0 \quad (5)$$

where

$$a=1+W_1^2+W_3^2$$

$$b=K_{11}W_1+2W_1W_2+K_{13}+2W_3W_4+K_{12}W_3$$

$$c=K_{11}W_2+W_4^2+W_2^2+K_{14}+K_{12}W_4$$

Equation (5) indicates that moving platform can reach two different poses. Finally, the center of moving platform P is calculated using (6) as follows

$$P=(X+r/3\,(cos(\theta_2)+cos(\theta_3))\ \ Y\ \ Z+r/3\,(sin(\theta_2)+sin(\theta_3))\ (6)$$

B. Velocity analysis

Velocity analysis involves determining the velocity state of the moving platform with respect to the fixed platform [29], when the joint rates $\dot{\alpha}_{4i}=\dot{q}_{4i}$ and $\dot{\alpha}_i=\dot{q}_i$ are given.

Let $\mathbf{V}=[0,\mathbf{v}]^T$ be the velocity state of the moving platform as observed form the fixed platform, where **v** is the velocity vector of any point on the moving platform. The velocity state **V** may be expressed in screw form through the limbs of the mechanism as follows

$$\dot{q}_{4i}\,{}^0\$_i^1+\dot{q}_i\,{}^1\$_i^2+{}_2\omega_3^i\,{}^2\$_i^3+{}_3\omega_4^i\,{}^3\$_i^4+{}_4\omega_5^i\,{}^4\$_i^5+{}_5\omega_6^i\,{}^5\$_i^6=\mathbf{V} \qquad (7)$$

therein i denote the i-th kinematic chain. In order to obtain an input-output velocity equation of the reconfigurable manipulator, consider that $L_i=[lx_i\ ly_i\ lz_i\ Lx_i\ Ly_i\ Lz_i]$ is a line in Plücker coordinates, which passes through the points B_i and C_i (Fig. 3), reciprocal to all the screws in the same limb, excepting the screws associated to the active joints. Therefore, the systematic application of the Klein form, which is denoted as $\{*;*\}$ of the line L_i to both sides of (7) and reducing terms, yields the input-output velocity equation as

$$\mathbf{v}=\mathbf{A}^{-1}(\mathbf{B}'\dot{\mathbf{q}}_4+\mathbf{B}\dot{\mathbf{q}}) \qquad (8)$$

therein $\mathbf{A}=[lx_i\ ly_i\ lz_i]^T$ is the manipulator Jacobian matrix, while $\dot{\mathbf{q}}_4=[\dot{q}_{41}\ \ \dot{q}_{42}\ \ \dot{q}_{43}]^T$ and $\dot{\mathbf{q}}=[\dot{q}_1\ \ \dot{q}_2\ \ \dot{q}_3]^T$ are the first-order generalized coordinates of the mechanism. The passive Jacobian are $\mathbf{B}'=diag[\{{}^0\$_1^1;L_1\}\ \ \{{}^0\$_2^1;L_2\}\ \ \{{}^0\$_3^1;L_3\}]$ and $\mathbf{B}=diag[\{{}^1\$_1^2;L_1\}\ \ \{{}^1\$_2^2;L_2\}\ \ \{{}^1\$_3^2;L_3\}]$.

C. Evident singular configurations

The singular configurations or singularities according to Refs. [30] and [31], can be deduced from (8). Such configurations are distinct for the fixed platform and the rest of the mechanism, since they are uncoupled mechanisms. However, when there exist combined singularities between these two mechanisms, then the reconfigurable manipulator will be in a fully singular pose.

In this concern, singularities of the Delta-type robot occurs when:

1. If $det(\mathbf{A})=0$, then the links Lb are collinear or parallels. In these configurations the moving platform gains degrees of freedom.

2. If $det(\mathbf{B})=0$, then the links La are completely folded or unfolded. In such configurations the mobile platform loses degrees of freedom.

3. If $det(\mathbf{A})=0$ and $det(\mathbf{B})=0$, then the fixed platform and the moving platform are coplanar, e.g. when the moving platform has an identical geometry to that of the fixed platform and the limbs are vertical.

It should be noted that the singularity locus remains the same as a one Delta-type robot. This condition shows that the reconfiguration strategy does not drastically affects the structure of this parallel robot. It is also important to note that the Delta-type mechanism is capable to avoid singularities by using the redundancy of the reconfigurable robot, which is an additional advantage of the proposed strategy in this paper.

On the other hand, the fixed platform that is reconfigurable, has its own singular configurations, which may occur when $det(\mathbf{B}')=0$, e.g. when the mobile elements are folded or unfolded.

IV. NUMERICAL EXAMPLE

In this section, a numerical example is provided to show that by redundancy it is possible to obtain customized solutions of the kinematic performance. For this aim, the compute of the condition number is used as a performance index throughout defined trajectories (See Fig. 5).

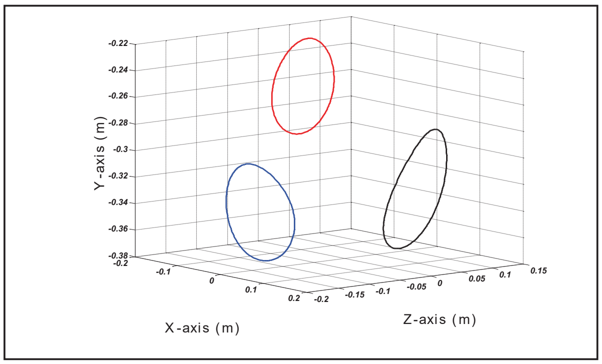
Three Cartesian trajectories of the center P of the moving platform are generated by solving the direct displacement analysis [29], considering $\alpha_{1,3}(t)=0.25\sin(t)$ and $\alpha_2(t)=0.25\cos(t)$ with an interval for time t of $0\le t\le 2\pi$. The configurations of α_{4i} for each trajectory are arbitrarily selected, shown in Table 1 as follows

TABLE I. ANGULAR CONFIGURATION FOR EACH TRAJECTORY

Trajectory 1 (blue color)	α_{41}= -15º	α_{42}= 30º	α_{42}= -70º
Trajectory 2 (red color)	α_{41}= -35º	α_{42}= 40º	α_{42}= 60º
Trajectory 3 (black color)	α_{41}= 45º	α_{42}= -45º	α_{42}= -25º

The link lengths used are: La =0.2, Lb =0.4, r =0.05, R_f=0.05 and $0 \leq R_r \leq 0.1618$, thanks to the reconfigurations effects. The direction of rotation axes of the active joints are defined by $\hat{u}_1 = -\hat{k}$, $\hat{u}_2$ = -0.866$\hat{\imath}$ +0.5$\hat{k}$ and $\hat{u}_3$ = 0.866$\hat{\imath}$ +0.5$\hat{k}$.

Figure 4. Trajectories of the center P at different values of α_{41}



The trajectories in Fig. 4 are performed in different planes and are considering the three degrees of freedom of the mechanism.

V. Reconfiguration effects on the robot performance

In [32] is defined that the condition number describes the accuracy/dexterity and the closeness to a singular configuration of a parallel manipulator. For this reason, many authors (as some of those aforementioned) have used the condition number as a performance index of a Delta-type robot. In our case, the results of this performance index are improved by using the redundancy generated by the proposed reconfiguration. The condition number is defined as

$$k = \left\| \mathbf{J}^{-1} \right\| \left\| \mathbf{J} \right\| \quad (9)$$

where **J** is the manipulator Jacobian matrix, obtained from Eq. (4) as follows

$$\mathbf{v} = \left(\mathbf{A}^{-1}\mathbf{B}\right)\dot{\mathbf{q}} = \mathbf{J}\dot{\mathbf{q}} \quad (10)$$

since we here use the matrix **J** of the well-known architecture of the Delta-type robot, in order to demonstrate that it is possible to enhance its kinematic performance thanks to the redundancy of the reconfigurable mechanism.

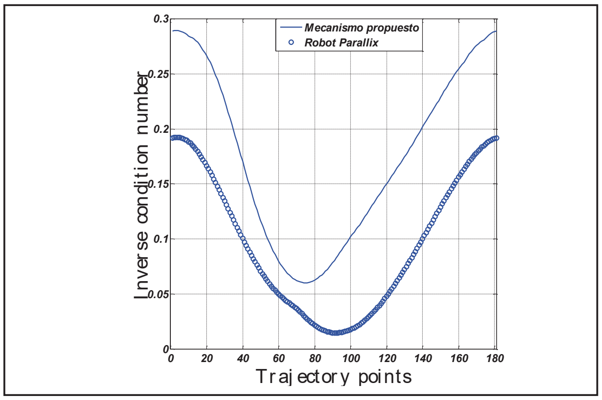
It is important to define that the first term of Eq. (8) depends on $\dot{q}_{4i}$, as well as $k(\alpha_{4i})$ given that the best solution of k is computationally obtained by sweeping all possible combinations of α_{4i} in a range of $90° \leq \alpha_{4i} \leq -90°$ and seeking locally for each one of the trajectory points. Therefore the first term of Eq. (8) vanishes.

To have a better appreciation of the condition number, the inverse k^{-1} will be used, since it has a range between $0 \leq k^{-1} \leq 1$. If k^{-1} = 0 indicates that the Jacobian matrix is not invertible, hence the manipulator cannot generate velocities in some directions which is called, to be at a singular configuration [31]. When k^{-1} = 1 the manipulator is capable of generating velocities in any direction, in other words to be in an isotropic pose [32].

The best results of k^{-1} are obtained for the 181 points of each trajectory by using the redundancy of the reconfigurable robot and they are compared with the result of k^{-1} obtained with the original configuration of the Delta-type robot (see Fig. 5 – Fig. 7).

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Figure 5. Values of k^{-1} for the first trajectory (blue circle) of Fig. 5

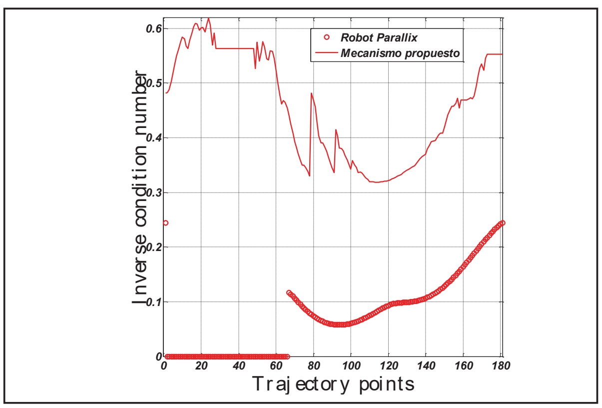


Figures 5 – 7 show a similar behavior of the kinematic performance of the two mechanisms in each of the trajectories. Nonetheless, the condition number of the Parallix robot is almost always less than 0.2 in each trajectory.

It is evident that in Fig. 5, the kinematic performance of Delta-type robot is slightly improved from 0.2 – 0.3 according to the condition number. However, along the entire trajectory, the reconfiguration strategy is better than the original configuration of the Delta-type robot.

An interesting case in the red trajectory is shown in Fig. 6. Note that from trajectory points 2 – 66, the Delta-type robot cannot perform its trajectory. In other words, along this section this robot would be in a singularity. On the other hand, the reconfigurable mechanism can easily escape from such singularities, therefore its performance is better than Delta-type robot along the entire trajectory. It should be noted that the improvement is highly significant, thanks to redundancy generated by the proposed reconfiguration strategy.

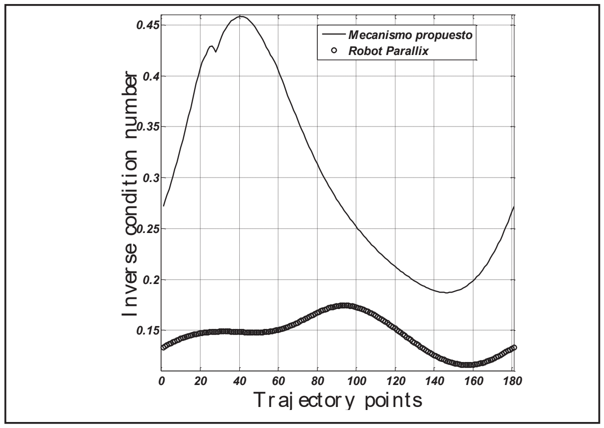
Figure 6. Values of k^{-1} for the second trajectory (red circle) of Fig. 5



It is easy to notice that in Fig. 7, a very important improvement occurs in the performance between the black trajectory points 20 – 60, since the reconfigurable robot reaches a maximum value $k^{-1} = 0.4583$ compared to the original mechanism that reaches a maximum one $k^{-1} = 0.1742$.

In each point of the prescribed trajectories, the reconfigurable mechanism obtains values of k^{-1} nearest to 1, which could not be achieved only with the Delta-type robot, namely when $\alpha_{4i} = 0$.

Figure 7. Values of k^{-1} of the third trajectory (black circle) of Fig. 5



Therefore, since the results of the condition number are improved in each trajectory thanks to the reconfiguration, all the other performance indices that depend on the manipulator Jacobian matrix, e.g.: Manipulability index, Minimum singular value, Global conditioning index, Global manipulability index and even the stiffness, are also improved.

In fact, the reconfiguration proposal can be used in flexible manufacturing systems, since it is considered to be able to customize its kinematic performance according to a specific task.

VI. Conclusion

Many reconfiguration strategies of parallel manipulators have been proposed in the robotic area (as aforementioned). However, the reconfigurability in parallel robots is still a challenge nowadays, since most of such reconfiguration strategies have a high level of mechanical complexity in their structures and hence they are very expensive to be made. In addition, a very limited number of authors have proposed reconfigurable Delta robots.

In this paper, a new reconfiguration of a Delta-type parallel robot is proposed. This reconfiguration strategy proposed here is highly versatile, very simple and it can be able to return to the original configuration and mechanical characteristics of a Delta-type robot.

The reconfiguration proposal, which is now in patenting process at IMPI-Mexico, consists of a fixed platform comprised by a fixed element and mobile elements, which simultaneously can modify the dimensions and shape of such fixed platform.

It is very important to note that it is could be considered that the use of the 3 additional actuators (3a, 3b, 3c in Fig. 2) in the reconfiguration proposal, is an inefficient and expensive application. Nevertheless, thanks to the reconfiguration proposal, significant advantages can be obtained with respect to a Delta-type parallel robot.

The kinematic analysis is reported by resorting to reciprocal-screw theory and the evident singular configurations are also analyzed. Customized solutions of the robot kinematic performance can be obtained by using the redundancy produced by the reconfiguration. Additional advantages such as: increased workspace and to avoid singular configurations, can be obtained.

A numerical example of effects of the reconfiguration on the manipulator performance is provided. The computation of the best condition number is obtained, according to prescribed trajectories to different heights of the robot workspace. The results show that by using the redundancy, the condition number is enhanced in each trajectory. Accordingly, all the performance indices that depend on the Jacobian matrix are also improved.

References

[1] M. Krefft, et al., "A reconfigurable parallel robot with high flexibility". In: Proceeding of ISR Robotik 2006 - Joint Conference Robotics München.
[2] L. J. Du Plessis, J. A. Snymar, "An optimally reconfigurable planar Gough-Stewart machining platform", *Mechanism and Machine Theory,* Vol. 41, 2006, pp. 334-357.
[3] S. Kumar, et al., "Characterization of reconfigurable Stewart platform for contour generation". *Robotics and Computer-Integrated Manufacturing*, Vol. 25, 2009, pp. 721-731.
[4] C.-T Chen, "Reconfiguration of a parallel kinematic manipulator for the maximum dynamic load-carrying capacity". *Mechanism and Machine Theory*, Vol. 54, 2012, pp. 62-75.
[5] J. Zhiming, S. Phillip, "Design of a reconfigurable platform manipulator", *Journal of Robotic Systems*, Vol. 15, No. 6, 1998, pp. 341-346.
[6] G. Gogu, "Isogliden-TaRb: a family of up to five axes reconfigurable and maximally regular parallel kinematic machines". *Journal Manufacturing Systems*, Vol. 36, No. 5, 2007, pp. 419-426.
[7] N. Plitea, et al., "Structural design and kinematics of a new parallel reconfigurable robot". *Robotics and Computer-Integrated Manufacturing*, Vol. 29, 2013, pp. 219-235.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

[8] L. Carbonari, et al., "A new class of reconfigurable parallel kinematic machines". *Mechanism and Machine Theory*, Vol. 79, No. 2, 2014, pp. 173-183.
[9] N. Simaan, M. Shoham, "Stiffness synthesis of a variable geometry six-degrees-of-freedom double planar parallel robot". *The International Journal of Robotics Research*, Vol. 22, No. 9, 2003, pp. 757-775.
[10] J. Bòrras, et al., "A reconfigurable 5-DoF 5-SPU parallel platform." *ASME/IFToMM International Conference on Reconfigurable Mechanism and Robots*, 2009, pp. 617-662.
[11] D. Zhang, Q. Shi, "Novel design and analysis of a reconfigurable parallel manipulator using variable geometry approach". *Practical Applications of Intelligent Systems*. *Advances in Intelligent and Soft Computing*, Vol. 124, 2012, pp. 447-457.
[12] W. Ye, et al., "A new family of reconfigurable parallel mechanism with diamond kinematotropic chain". *Mechanism and Machine Theory*, Vol. 74, 2014, pp. 1-9.
[13] S. Kock, W. Schumacher, A parallel x-y manipulator with actuation redundancy for high-speed and active-stiffness applications. *Robotics and Automation 1998*. *Proceedings 1998 IEEE International Conference on*, Vol. 3, pp. 2295-2300.
[14] Z. M. Bi, B. Kang, Enhancement of adaptability of parallel kinematic machines with an adjustable platform., *Journal of Manufacturing Science and Engineering*, Vol. 132, No. 6, 2010.
[15] A. Moosavian, F. (J). Xi, Design and analysis of reconfigurable parallel robots with enhanced stiffness. *Mechanism and Machine Theory*, Vol. 77, 2014, pp. 92-110.
[16] Y. Jiang, T-m. Li, L-p. Wang, Dynamic modeling and redundant force optimization of a 2-DOF parallel kinematic machine with kinematic redundancy. *Robotics and Computer-Integrated Manufacturing*, Vol. 32, 2015, pp. 1-10.
[17] C. Li, X.G. Zhang, Y. S. Zhao, Redundant actuation investigation of a 5-DOF parallel robot. *Applied Mechanics and Materials*, Vol. 526, 2014, pp. 164-170.
[18] J. Kotlarski, B. Heimann, T. Ortmaier, Improving the pose accuracy of planar robot using mechanism of variable geometry. *Advances in Robot Manipulators*, *InTech*, 2010.
[19] J. Kotlarski, B. Heimann, T. Ortmaier, Influence of kinematic redundancy on the singularity-free workspace of parallel kinematics machines., *Frontiers of Mechanical Engineering*, Vol. 2, No. 2, 2012, pp. 120-134.
[20] S-H. Cha, T. A. Lasky, S. A. Velinsky, Singularity avoidance for the 3-RRR mechanism using kinematic redundancy. *IEEE International Conference on Robotics and Automation*, 2007, pp. 10-14.
[21] B. Xu, T. Li, X. Liu, J. Wu, Workspace Analysis of the 4RRR Planar Parallel Manipulator with Actuation Redundancy. *Tsinghu Science and Technology*, Vol. 15, No. 5, 2010, pp. 509-516.
[22] S. Yue, S. K. Tso, W. L. Xu, Maximum-dynamic payload trajectory for flexible robot manipulators with kinematic redundancy. *Mechanism and Machine Theory*, Vol. 36, 2001, pp. 785-800.
[23] J. Wu, J. Wang, L. Wang, Optimal kinematic design and application of a redundantly actuated 3DOF planar parallel manipulator. *Journal of Mechanical Design*, Vol. 130, No. 5, 2008.
[24] A. D. Finistuari, F. (J). Xi, Reconfiguration analysis of a fully reconfigurable parallel robot. *Journal of Mechanism and Robotics*, Vol. 5, No. 4, 2013.
[25] W. Dan, F. Rui, C. Wuyi, Performance enhancement of a three-degree-of-freedom parallel tool head via actuation redundancy. *Mechanism and Machine Theory*, Vol. 71, 2014, pp. 142-162.
[26] Y. Jiang, T. Li, T. Wang, The dynamic modelling, redundant-force optimization, and dynamic performance analyses of a parallel kinematic machine with actuation redundancy. *Robotica*, Vol. 33, 2015, pp. 241-263.
[27] A. Gonzalez-Hernandez. E. Castillo-Castaneda, "Stiffness estimation of a parallel manipulator using image analysis and camera calibration techniques". *Robotica*, Vol. 31, 2013, pp. 657-667.
[28] A. Balmaceda, E. Castillo, "Metodología de rediseño de un robot paralelo tipo Delta de 3-GDL en función de un espacio de trabajo prescrito". (In spanish), *Nexo, Revista Científica*, Vol. 26, No. 01, 2013, pp. 24-33.
[29] J. Gallardo-Alvarado, A. L. Balmaceda-Santamaría, E. Castillo-Castaneda. "An application of screw theory to the kinematic analysis of a Delta-type robot". *Journal of Mechanical Science and Technology*, Vol. 28, No. 9, 2014, pp. 3785-3792.
[30] C. Gosselin, J. Angeles, Singularity analysis of closed-loop kinematic chains. *IEEE, Transactions on Robotics and Automation*, Vol. 6, No. 3, 1990, pp. 281-290.
[31] L.W. Tsai, *Robot Analysis: The mechanics of serial and parallel manipulators*. A Wiley-interscience publication, United States of America, 1999.
[32] J. P. Merlet, Jacobian, manipulability, condition number and accuracy of parallel robots. *Robotics Research*. *Springer Tracts in Advanced robotics*, Vol. 8, 2007, pp. 175-184.
[33] A. L. Balmaceda-Santamaría, E. Castillo-Castaneda, J. Gallardo-Alvarado. A novel reconfiguration strategy of a Delta-type parallel manipulator. International Journal of Advanced Robotic Systems, Vol. 13, 2016.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 5, páginas 41 - 50.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Robot bípedo antropométrico con dieciocho grados de libertad. Enfoque mecánico.

D. Alvarado Rivera, L. G Corona Ramírez, J. S. Muñoz Reina

***Resumen*: El objetivo principal de este trabajo es presentar; el análisis estructural de la construcción de un prototipo de robot humanoide, la longitud de los eslabones y la ubicación de las articulaciones basándose en la antropometría del cuerpo humano, el diseño de los eslabones se realiza en función del análisis de esfuerzos que se presentan en la composición del robot y el par al que se someten los mismos, como punto final se realiza un análisis cinemático utilizando la metodología de Denavit-Hartenberg, con el fin de conocer la posición y orientación de los eslabones que conforman al robot.**

I. Introducción

La robótica en nuestro tiempo ha tenido un gran avance en los recientes años, es por ello que las nuevas generaciones buscan desarrollar metodologías para la creación de robots, en su mayoría a estos se les atribuyen tareas de servicio las cuales conllevan el riesgo de la vida de un individuo.

Este trabajo presenta, el diseño de un robot bípedo por medio de análisis de esfuerzos, el cual se comunica con un sistema externo que mide y envía la posición de los ángulos existentes en las articulaciones de un individuo durante su marcha. Uno de los principales retos es obtener un diseño funcional de un robot bípedo que cuente con 18 grados de libertad.

Al gestionar este tipo de robots principalmente se busca una adecuada morfología, para esto se determinó la longitud, la composición estructural de cada elemento que compone al robot y los ángulos de trabajo de los elementos que dan sostén a cada articulación, asemejando así la constitución anatómica del cuerpo humano desde los pies hasta los brazos.

Utilizando los parámetros Denavit-Hartenberg es posible predecir el comportamiento cinemático del bípedo debido a que se obtienen las trayectorias y orientación final de los eslabones del robot, con estos datos se obtiene la relación existente entre coordenadas articulares y cartesianas.

II. Articulaciones presentes en el robot

Usualmente los bípedos no presentan los mismos movimientos en comparación a los de un ser humano, basta con mencionar que la articulación de la mano podría modelarse como un robot esférico, de tal manera para imitar la gran mayoría de los movimientos que realiza la muñeca serían necesarios tres motores encargados de generar estos movimientos. Quizá para un robot da tamaño real no causaría problema utilizar esos actuadores, pero al llevarlo a menor escala la dificultad aumenta considerablemente, por lo que para este diseño del humanoide se busca que cuente con las articulaciones más representativas del cuerpo humano.

Tomando en cuenta lo anterior podemos destacar los movimientos más notorios al caminar, como son la oscilación que existe en los brazos, la inclinación del dorso, y la convergencia de las actividades del miembro inferior. La marcha humana se divide en dos fases repetitivas; la primera es la fase portante el inicio del clico, que a su vez se divide en tres periodos 1. Apoyo de los dos miembros inferiores, 2. Apoyo de un solo miembro y 3. Apoyo de los dos miembros inferiores, y la segunda fase denominada oscilante, en la **Fig. 1,** se aprecian las dos etapas, así como el porcentaje que cada una representa.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

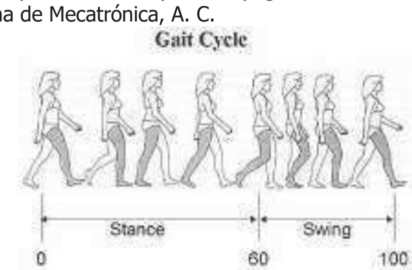


Figura.1. Ciclo de la marcha del humano [1]

III. Biómecanica de las articulaciones

La estructura mecánica del cuerpo humano se analiza como una serie eslabones interconectados, cuyas uniones generan movimientos en los miembros del ser humano, de esta manera se presenta en la **Tabla 1 y 2** los rangos angulares utilizados para el diseño del robot bípedo.

Tabla 1: Articulaciones presentadas en el plano frontal y sagital

Rangos de Movimiento en los planos frontal y sagital		
Articulación	Movimiento	Rango
Cuello	Rotación	0°-180°
Dorso	Inclinación	0°-40°
Hombro	Flexión	0°-180°
	Extensión	0°-50°
	Aducción	0°-30°
	Abducción	0°-180°
Codo	Flexión	0°-145°
Cadera	Flexión	0°-120°
	Extensión	0°-15°
	Aducción	0°-20°
	Abducción	0°-30°
Rodilla	Flexión	0°-120°
	Extensión	0°-5°
Tobillo	Flexión	0°-40°
	Extensión	0°-20°

Tabla 2: Articulaciones presentadas en el plano transversal

Rangos de Movimiento en el plano transversal		
Articulación	**Movimiento**	**Rango**
Tobillo	Eversión	0°-30°
	Inversión	0°-35°

Con los valores mostrados en las tablas anteriores se consideran como parámetros de diseño para el robot de 18 grados de libertad, además de aseverar que los actuadores empleados en este trabajo serán servomotores, debido a su rango de trabajo (usualmente de 0°-180°).

IV. Antropometría del ser humano

La antropometría se dedica a la determinación o estandarización de las medidas longitudinales que existen en el cuerpo de un individuo, con este tipo de estudios se puede establecer una escala determinada para cada segmento del cuerpo humano, teniendo como base la altura de este, en la **Fig. 2**, se aprecia cómo los valores de distancia están acotados por la altura de una persona.

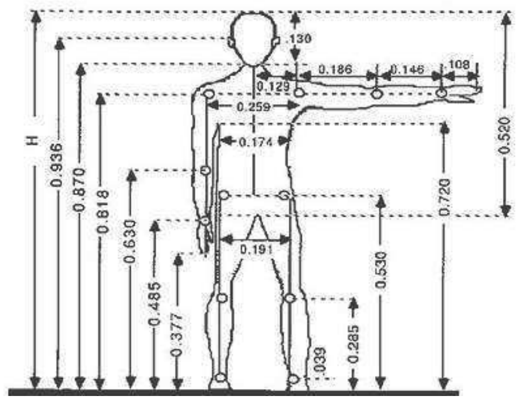


Figura.2. Antropometría del cuerpo humano [2]

De acuerdo a la antropometría del cuerpo humano **Fig. 2**, se establece una altura de 43 cm, para el robot bípedo, utilizando ésta se obtienen las longitudes de los eslabones que constituirán al bípedo, en la **Tabla 3**, se muestran dichos parámetros.

Tabla 3: Segmentos establecidos por la antropometría

Segmento	Distancia (cm)
Pie-Tobillo	2.65
Tobillo-Rodilla	8.435
Rodilla-Cadera	8.6
Cadera-Dorso	7.7
Dorso-Hombro	9.15
Hombro-Brazo	1.92
Brazo-Antebrazo	6.595
Dorso-Cabeza	6.52

V. Construcción del robot bipedo.

A. Generación de las piezas estructurales.

La consideración establecida para la construcción de la estructura del bípedo se basa en la anatomía del miembro inferior y superior del cuerpo humano. Para ello se realizó un modelo en CAD (por sus siglas en inglés Computer Aided Design). La morfología de los elementos estructurales fue concebida para que se adaptaran a las características de los actuadores (TowerPro® MG995 [3]). El robot esta constiruido por 49 piezas, las cuales brindan soporte a los 18 actuadores que a su vez inducen los movimientos que realiza el bípedo.

El robot presenta caracteristicas especiales en cuanto a la imitacion de los movimientos del cuerpo humano como son la flexión, extensión, de la cadera, rodilla, codo y hombro, aducción y abducción, de la cadera y hombro, la pronación y supinación de los tobillos, la inclinación del dorso y la rotación de la cabeza.

En la **Fig. 3** se presenta el ensamblaje final del robot. Las cadenas cinemáticas presentan un eslabón inicial o base(cadera), posteriormente la siguen las conexiones sucesivas de eslabones hasta llegar al efector final. Para este caso el efector final del el miembro superior son los brazos, en el caso del miembro inferior son los pies los cuales brindan el equilibrio necesario al robot.

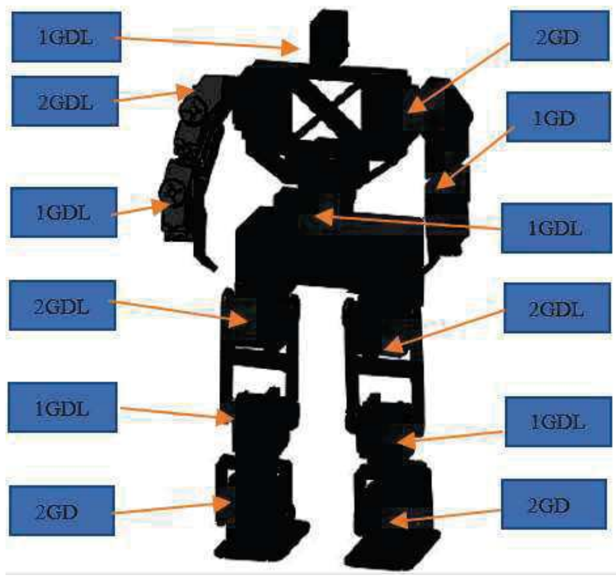


Figura 3. Ubicación de los grados de libertad (GDL) del robot.

Las articulaciones que conforman al robot son las encargadas de transmitir la energía y producir movimiento, así como las trayectorias a cada uno de los eslabones que conforman el robot. Para lo cual se emplean 18 articulaciones rotacionales (servomotor MG995) y su distribución se presenta en la **Fig. 3**.

B. Simulación de análisis de esfuerzos

Una vez generado el modelo en CAD se realiza una simulación de análisis de esfuerzos verificando así los parámetros de diseño establecidos, en donde se midió la deformación y el punto de ruptura de cada elemento estructural. Estos conceptos se generan con la asistencia del software COMSOL 5.1™, cada parte del robot se somete a una carga igual al doble del peso teórico del bípedo de 30N. El material utilizado fue el Acrilonitrilo Butadieno Estireno, debido a que presentó las propiedades mecánicas (mostradas en la **Tabla 4** [7]) adecuadas para la manufactura de los eslabones

TABLA 4: PROPIEDADES MECÁNICAS DE ABS

Propiedades mecánicas del ABS	**Valor**	**Propiedades mecánicas del ABS**	**Valor**
Punto de ruptura (psi)	2500-800	Fuerza de flexión (psi)	4000-14000
Módulo de tensión (x10^3 psi)	130-420	Dureza (Rockwell)	R75-115
Elongación a la ruptura (%)	20-100	Coeficiente de expansión térmico (10^{-6} in/°F)	60-130
Fuerza compresiva (psi)	5200-10000	Temperatura máxima de operación (°C)	35
Módulo de flexión (x10^3)	130-440		

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

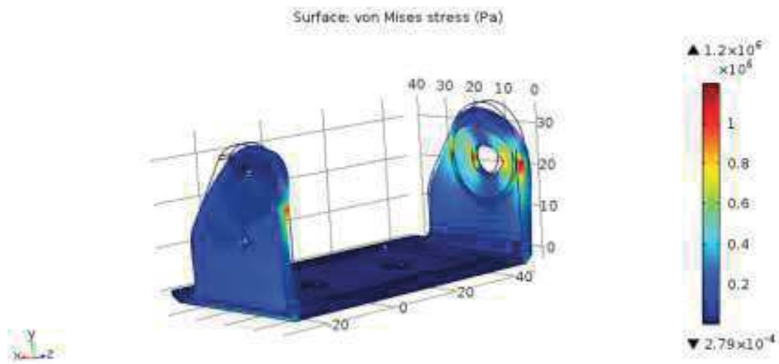


Figura 4. Simulación de esfuerzos en el elemento “pie”.

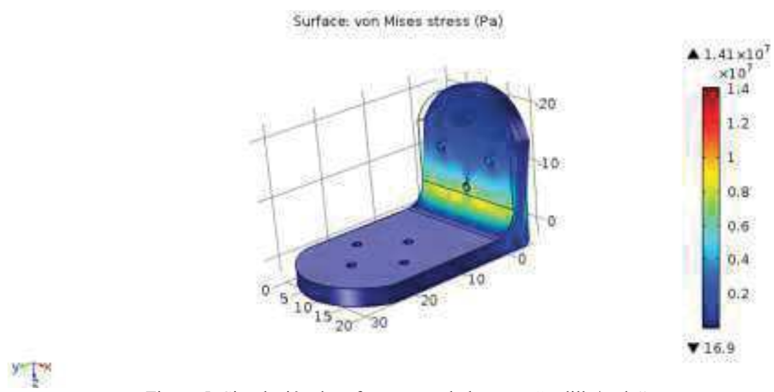


Figura 5. Simulación de esfuerzos en el elemento “rodilla/codo”.

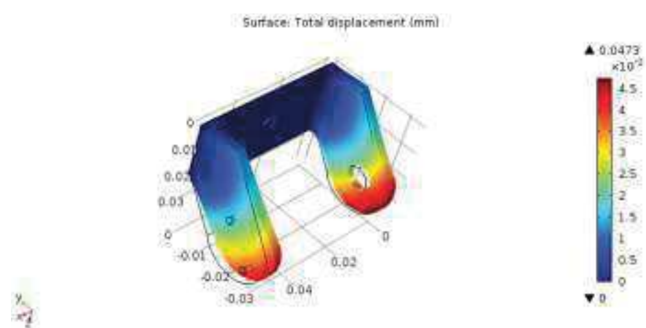


Figura 6. Simulación de esfuerzos en el elemento “brazo”.

Un ejemplo de los esfuerzos de von Mises se observa en la **Fig. 4, 5 y 6,** esta información y la deformación máxima que sufren los elementos, en la gráfica lateral se ejemplifican los rangos mínimos y máximos para cada parámetro considerado en la simulación los cuales indican si los eslabones y el material cumplen con el rendimiento requerido para la dinámica del robot, los resultados se muestran en la **Tabla 4**. Cuyos datos están relacionados a las ecuaciones (1) y (2) respectivamente.

$$\sigma_{vM} = \sqrt{\frac{(\sigma_1-\sigma_2)^2+(\sigma_2-\sigma_3)^2+(\sigma_3-\sigma_1)^2}{2}} \qquad (1)$$

$$N = \frac{\sigma_m}{\sigma_{VM}} \geq 1 \qquad (2)$$

Donde: σ_1, σ_2 y σ_3, son los mayores esfuerzos que actúan sobre el elemento al aplicar una fuerza, estos se obtienen a partir del tensor de esfuerzos [4]. N (o factor de seguridad), es el cociente entre el valor otorgado por los esfuerzos de von Mises (σ_{vM}) y el valor máximo de la resistencia a la tracción del material (σ_m).

En la **Tabla 5**, se identifica que el factor de seguridad de cada elemento siempre es mayor a uno lo que nos asegura analíticamente que el peso del robot no condicionara la estructura mecánica.

TABLA 5: RESULTADOS DEL ANÁLISIS DE ESFUERZOS

PIEZA	Esfuerzos de Von Mises (Pa)	Desplazamiento total (m)	Factor de seguridad
Base Pie	1.14E+04	3.43E-09	3.73E+03
Pie	1.20E+06	1.81E-05	3.54E+01
Pierna/Muslo	2.04E+06	1.32E-05	2.08E+01
Rodilla/Codo	1.41E+07	3.88E-04	3.01E+00
Carcaza 1	1.35E+07	4.43E-04	3.15E+00
Carcaza 2	1.71E+07	3.98E-04	2.49E+00
Carcaza 3	1.13E+07	3.21E-04	3.76E+00
Carcaza 4	1.10E+07	3.15E-04	3.86E+00
Cadera	6.89E+06	2.90E-04	6.17E+00
Dorso	1.00E+07	4.83E-05	4.25E+00
Brazo	6.42E+06	4.73E-05	6.62E+00
Antebrazo	1.93E+06	8.86E-06	2.20E+01
Mano	2.23E+07	4.32E-04	1.91E+00
Contraeje	5.75E+06	8.48E-06	7.39E+00

Los motores TowerPro® MG995 encargados del movimiento de las juntas del robot, se adoptaron en base al par ejercido sobre el centro de cada servomotor. Utilizando la Ecuación (3). Dónde: τ es el momento de fuerza, F es la fuerza que se aplica en un elemento en este caso es el peso, y d es la longitud del brazo de palanca que va desde el eje del motor hasta el extremo final del eslabón, así se obtuvieron los pares máximos entre los eslabones los cuales se presentan en la **Tabla 5**.

$$\tau = F \cdot d \qquad (3)$$

Con la información mostrada en la **Tabla 6** y tomando en cuenta el par máximo de los actuadores de 1.471 $N \cdot m$ [3] se concluye que los servomotores son los adecuados para su uso en este prototipo.

TABLA 7: PAR MÁXIMO ENTRE ESLABONES.

Eslabones	Par máximo ($N \cdot m$)
Cadera y rodilla	0.3789
Cadera y tobillo	0.89123
Cadera y pie	1.04

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Cadera-Dorso	1.407
Hombro y mano	0.336
Antebrazo y mano	.249

VI. Modelo matemático del robot

La cinemática directa está enfocada al estudio analítico del movimiento del robot con respecto a un eje de referencia cartesiano despreciando las fuerzas que originan movimiento. Por otra parte, es una función vectorial que relaciona coordenadas articulaciones a coordenadas cartesianas $[x, y, z]^T \in R^3$, así como la orientación $[\theta, \emptyset, \varphi]^T \in R^3$ del extremo final de una cadena cinemática.

Para realizar este prototipo es necesario conocer el espacio de trabajo con el fin de predecir el comportamiento en el espacio de trabajo del robot para lo cual se toma como base cada uno de los ángulos de articulares en los planos frontal sagital y transversal de una persona promedio [1], estos ángulos se resumen en la **Tabla 1 y 2**.

Para la obtención de la cinemática del robot [5] se analizan 4 cadenas cinemáticas, 2 pertenecientes a los miembros superiores y 2 a los inferiores, el análisis se debe de analizar tomando como referencia la cadera (punto Or Figura 7) y analizar como una cadena cinemática abierta las extremidades del robot. Para obtener dicho modelo se emplea el algoritmo Denavit-Hartenberg [6]. En la **Fig. 7** se representan las juntas del robot y su separación.

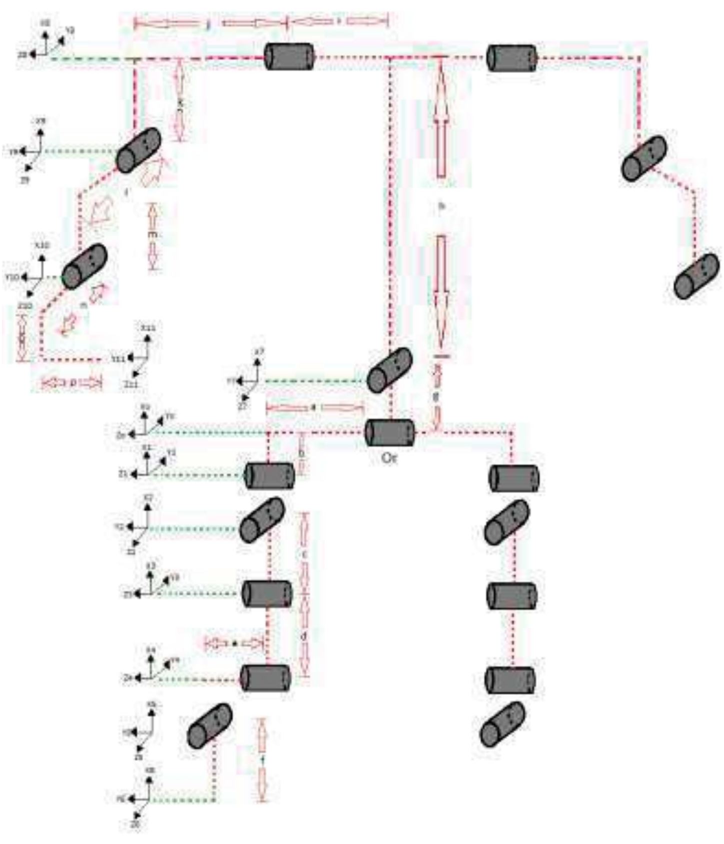


Figura 7. Sistema de ejes coordenados del robot.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

El método Denavit-Hartenberg consiste en determinar una tabla de paramentos θ, d, l y α relacionados con los eslabones del robot. En donde θ es el ángulo de rotación sobre el eje z ($R_{z,\theta}$), d es la distancia de translación sobre el eje z ($T_{z,d}$), l es la distancia de translación sobre el eje x ($T_{x,l}$) y α el ángulo de rotación sobre el eje x ($R_{x,\alpha}$).

Los parámetros Denavit-Hartenberg se representan por un producto de 4 matrices básicas:

$$H_i = R_{z,\theta_i} T_{z,d_i} T_{x,l_i} R_{x,\alpha_i} = \begin{bmatrix} \cos(\theta_i) & -sen(\theta_i) & 0 & 0 \\ sen(\theta_i) & \cos(\theta_i) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & l_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & \cos(\alpha_i) & -sen(\alpha_i) & 0 \\ 0 & sen(\alpha_i) & \cos(\alpha_i) & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (4)$$

Los parámetros de Denavit-Hartenberg que modelan el comportamiento cinemático del robot se presentan en la **Tabla 6** y **Tabla 7**.

TABLA 7: PARÁMETROS DE DENAVIT-HARTENBERG DE PIERNAS DEL ROBOT

	Pierna derecha				Pierna izquierda			
Eslabón	**θ**	**d**	**l**	**α**	**θ**	**d**	**l**	**α**
1	0	b	-a	0	0	-b	-a	0
2	θ_1	0	0	90	θ_1	0	0	-90
3	θ_2	0	-c	-90	θ_2	0	-c	90
4	θ_3	0	-d	0	θ_3	0	-d	0
5	θ_4	e	0	90	θ_4	-e	0	-90
6	θ_5	0	-f	0	θ_5	0	-f	0

TABLA 8: PARÁMETROS DE DENAVIT-HARTENBERG DE BRAZOS DEL ROBOT

	Brazo derecho				Brazo izquierdo			
Eslabón	**θ**	**d**	**L**	**α**	**θ**	**d**	**l**	**α**
1	0	0	-g	90	0	0	-g	90
2	θ_6	0	h	-90	θ_6	0	h	-90
3	θ_7	i+j	-k	90	θ_7	-i-j	-k	-90
4	θ_8	l	-m	0	θ_8	L	-m	0
5	θ_9	n	-o-p	0	θ_9	n	-o+p	0

De las **Tablas 7 y 8** empleando el software Matlab™ 2015 y aplicando las matrices de trasformación homogéneas (ecuación 4), se simula el comportamiento cinemático del robot para determinar los puntos finales de cada cadena cinemática.

A continuación, se muestra gráficamente el conjunto de transformaciones homogéneas, en la **Fig. 8**, se presenta el estado inicial del robot.

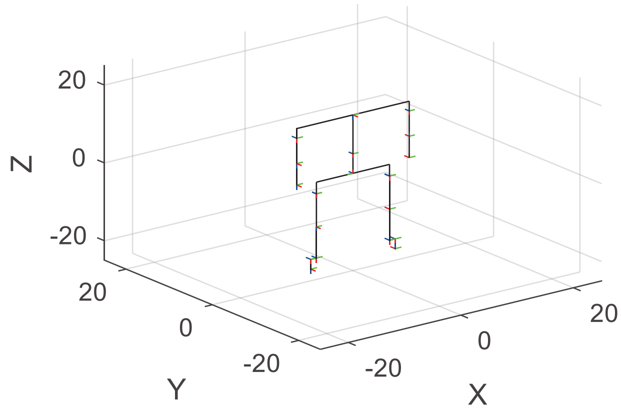


Figura 8. Sistema de ejes coordenados del robot.

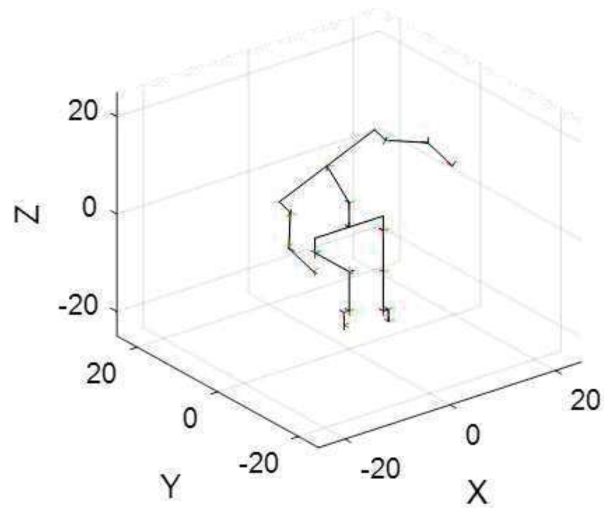


Figura 9. Representación gráfica del robot aplicando variaciones en articulaciones.

En la **Fig. 9**, se proponen variaciones angulares respecto al estado inicial, las cuales representan la flexión de 90º del muslo derecho, extensión de 90º de la pierna derecha, flexión lateral de 25º del dorso, flexión de 90º de brazos y antebrazos derecho e izquierdo, y abducción de 90 º de brazo derecho e izquierdo.

De la figura anterior se obtienen las matrices de transformación homogénea finales que representan las orientaciones y posiciones en el espacio cartesiano de los efectores finales (manos y plantas de los pies del robot) para este caso, las matrices resultantes en centímetros son:

- Brazo derecho

$$H_1 = \begin{bmatrix} 0 & -1 & 0 & -7.5 \\ 0 & 0 & -1 & -8.5 \\ 1 & 0 & 0 & -13.5 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Brazo izquierdo

$$H_2 = \begin{bmatrix} 0 & -1 & 0 & -7.5 \\ 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & -22 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

- Pierna derecha

$$H_3 = \begin{bmatrix} -0.29 & -0.9 & -0.29 & -13.68 \\ 0.7 & 0 & -0.7 & -8.9 \\ 0.64 & -0.42 & 0.64 & -0.1779 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Pierna izquierda

$$H_4 = \begin{bmatrix} -0.29 & 0.9 & 0.29 & 12.76 \\ 0.7 & 0 & 0.7 & -8.9 \\ 0.64 & 0.42 & -0.64 & 12.15 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

VII. CONCLUSIONES.

El robot construido cuenta dieciocho grados de libertad, cabe destacar que cada motor puede ser controlado independientemente o bien pueden trabajar de manera conjunta en dado caso de ser requerido. Además, presenta una altura de 43 cm y un peso de 1.5 kg, dichas características cumplen con los valores antropométricos mencionados.

Con las matrices de transformación homogénea es posible predecir la posición y orientación de cada eslabón en un momento en específico esto se realiza con el fin de evitar colisiones entre eslabones.

El método utilizado para analizar los esfuerzos involucrados en cada uno de los eslabones permito corregir de manera óptima la geometría de los eslabones, asegurando que la estructura del bípedo resistirá las fuerzas provocadas por el mismo, para un trabajo futuro se debe de considerar realizar un análisis dinámico más competo, con el fin de obtener por alguna metodología los centros de masa óptimos de cada eslabón, los cuales permitan brindarle un mejor balanceo y control al robot.

En proceso de elaboración, el material por el que se optó, redujo considerablemente los costos fabricación del bípedo. De ser necesaria una modificación (ya sea en la forma o el tamaño) de los elementos estructurales no existe inconveniente alguno puesto que se cuenta con esa información.

Considerando la información descrita aquí, para un trabajo a futuro se dará pie a un objetivo mayor que es la monitorización y detección de anomalías en el ciclo de marcha del ser humano, al fusionar este agente con un sistema capaz de detectar los ángulos de movimiento del cuerpo.

REFERENCIAS

[1] J. Rose and J. Gamble, Human walking. Philadelphia: Lippincott Williams & Wilkins, 2006.
[2] R. Drillis and R. Contini, Body segment parameters. New York, N.Y.: New York University, School of Engineering and Science, 1966.
[3] "Geelmed ingeniería y servicios", Geelmed.com. [Online]. Available:http://www.geelmed.com/documents/products/producto_000000132DOC.pdf?d=1312813326. [Obtenido: 04- mayo- 2015].
[4] S. Kalpakjian, S. Schmid and Espinoza Limón Jaime, Manufactura, ingenieria y tecnologia. México, D.F.: Pearson Educacion, 2008.
[5] Aguilar, D., & Vivas, A. (2009). Modelado y control de un robot bípedo de nueve grados de libertad. In VIII Congreso de la Asociación Colombiana de Automática.
[6] F. Reyes Cortes, Matlab aplicado a robotica y mecatronica. Alfaomega Grupo Editor, 2012.
[7] P. Schweitzer, Mechanical and corrosion-resistant properties of plastics and elastomers. New York: Marcel Dekker, 2000.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 6, páginas 51 - 58.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Sliding mode tracking control for an inertia wheel pendulum around its unstable open-loop equilibrium point

Luis T. Aguilar, Andhers N. Piña, and Daniel I. Aparicio

***Abstract*—In this paper we address a robust tracking control problem for an inertia wheel pendulum against uniformly bounded matched disturbances. The periodic motion of the pendulum will be at the up-right position which corresponds to the unstable equilibrium point of the unforced system. A two-relay controller based reference model was developed for generating the desired trajectories to be tracked by the unactuated link of the inertia wheel pendulum and the design of an sliding modes tracking controller. The desired amplitude and frequency were tuned by choosing the two-relay control gains properly. A sliding mode tracking controller was capable to track the prescribed reference trajectory. Performance issues of the synthesized controller were illustrated in a numerical study.**

I. Introduction

Control theory plays an important role in the accurate desired motion of all mechanical systems in spite of external disturbances, imprecision value of the physical parameters, nonmodelled and parasitic dynamics. Although the tracking control of fully-actuated systems are well studied, the tracking control for non-minimum phase systems is a challenging problem since that further research applications in the motion control of underactuated systems have gone in many directions. Few examples where underactuated systems required to be orbitally stabilized are biped robot under the walking phase [1, 2, 3], gymnastic robots, or electrical converters [4, 5].

The aim of this paper is to solve the tracking control problem for the inertia wheel pendulum under the presence of undesired external disturbances using sliding modes. The periodic motion of the pendulum will be at the upright position which corresponds to the unstable equilibrium point of the unforced system. The *two-relay controller* [6, 7, 8, 9] based reference model was used for generating the desired trajectories to be tracked by the unactuated link of the inertia wheel pendulum and then, design a robust tracking controller via sliding modes. The desired amplitude and frequency were set by choosing the two-relay control gains according to the formulas obtained from the describing function method [10]. This paper also addresses the finite-time stability proof of the closed-loop system.

The paper is organized as follows. In Section II, we give the dynamic model of the plant under study and the problem definition is also stated. In Section III, the trajectory generation, via two-relay controller, to be tracked by the output of the plant is revisited. In Section IV, a sliding mode tracking control is synthesized to guarantee robust tracking of the reference trajectory. Numerical results illustrate the performance of the proposed sliding modes controller in Section V. Finally, conclusions are provided in Section VI.

II. Dynamic Model and Problem Statement

Dynamics of an inertial wheel pendulum, taken from [11], augmented with viscous friction and disturbances is described as follows:

$$\begin{bmatrix} J_1 & J_2 \\ J_2 & J_2 \end{bmatrix} \begin{bmatrix} \dot{v}_1 \\ \dot{v}_2 \end{bmatrix} + \begin{bmatrix} h \sin q_1 \\ f_s\, v_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tau + w \tag{1}$$

$$y = [q_1, q_2]^T \tag{2}$$

where $q_1(t) \in \mathbb{R}$ is the absolute angle of the pendulum, $q_2(t) \in \mathbb{R}$ is the absolute angle of the disk, $v_1(t) = \dot{q}_1(t)$ and $v_2(t) = \dot{q}_2(t)$ are the angular velocity of the pendulum and the wheel, respectively; $t \in \mathbb{R}$ is the time, $y(t) \in \mathbb{R}^2$ is the measurement output, $\tau \in \mathbb{R}$ is the controlled torque applied to the wheel (see Fig. 1), $f_s v_2$ represents the viscous friction force affecting the actuator where $f_s > 0$ is the viscous friction coefficient, J_1 and J_2 are the inertia of the pendulum and that of the wheel, respectively; the factor $h = mgl$ depends on the combined mass m of the over-all wheel pendulum, on the gravitational constant g, and on the length l of the link, and finally, $w = (w_1, w_2)^T$ are the external disturbances affecting the system. An upper bound $W_i > 0$, (i = 1, 2) for the magnitude of the disturbances is normally known a priori

First author gratefully acknowledge the financial support from CONACYT under Grant 268364.
L. T. Aguilar is with Instituto Politécnico Nacional—CITEDI, Avenida Instituto Politécnico Nacional No. 1310 Colonia Nueva Tijuana, Tijuana 22435 México (email: laguilarb@ipn.mx).
A. Piña and D. Aparicio are with Instituto Politécnico Nacional—UPIITA, Avenida Instituto Politécnico Nacional No. 2580, Gustavo A. Madero, Barrio La Laguna, Ticomán, 07340, Ciudad de México (email: daparicio@ipn.mx).

$$\sup_t |w_i(t)| \ \leq W_i \quad i = 1, 2. \tag{3}$$

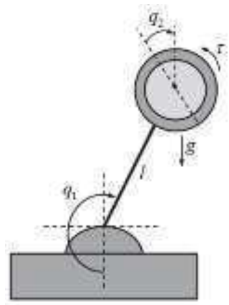


Fig. 1: Inertia wheel pendulum

The control objective for the underactuated system (1) is to design a control input τ to enforce the angular position of the pendulum q_1 to follow a desired periodic trajectory $q_{r1}(t)$, that is

$$\lim_{t\to\infty} \|q_{r1}(t) - q_1(t)\| = 0. \tag{4}$$

The desired trajectory is designed such that $q_r(t)$, $\dot{q}_r(t)$, and $\ddot{q}_r(t) \in \mathbb{R}^2$ are uniformly bounded in t.

III. Generating Desired Trajectories Centered at the Upright Equilibrium Point

Let us recall, from Iriarte *et al*. [12], the procedure for generating the set of trajectories for the inertia wheel pendulum to be tracked. First, let us consider the dynamics of the wheel pendulum in terms of the reference positions and velocities $(q_r(t), \dot{q}_r(t))$ without viscous friction

$$\begin{bmatrix} J_1 & J_2 \\ J_2 & J_2 \end{bmatrix}\begin{bmatrix} \ddot{q}_{1r} \\ \ddot{q}_{2r} \end{bmatrix} + \begin{bmatrix} h \sin q_{1r} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}\tau_r. \tag{5}$$

Next, we need to find the reference torque $\tau_r \in \mathbb{R}$ capable to produce a set of stable desired periodic motions at the non-actuated link $(y(t) = q_{1r}(t))$ such that the periodic motion has a desired frequency Ω and amplitude A_p. Throughout, we confine our research interest in desired oscillations around the upright position of the pendulum which correspond to a challenging problem.

The inertia wheel pendulum has underactuation degree one satisfying certain structural property provided in [13]. Thus, it is possible to make an exact linearization of (5) thus achieving local stability of zero dynamics, by taking the following change of variables

$$p_1 = q_{1r} + J_1^{-1}J_2q_{2r} - \pi \tag{6}$$
$$\eta = J_1\dot{q}_{1r} + J_2\dot{q}_{2r} + Kp_1 \tag{7}$$

where $K > 0$ is constant. The last term in (6) ensures that the desired trajectories will be generated around $q_{1r}^{\star} = \pi$ rad (see Fig. 1). From above set of equations, it is possible to verify that

$$J_1\dot{p}_1 = \eta - Kp_1$$

while

$$\begin{aligned} \dot{\eta} &= KJ_1^{-1}J_2\dot{q}_{2r} - h\sin(q_{1r}) + K\dot{q}_{1r}, \\ \ddot{\eta} &= -h\cos(q_{1r})\,\dot{q}_{1r} - KJ_1^{-1}h\sin(q_{1r}), \\ \dddot{\eta} &= R(q_{1r}, \dot{q}_{1r}) + H(q_{1r})\tau_r, \end{aligned}$$

where

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

$$H(q_{1r}) = \frac{h\cos(q_{1r})}{J_1 - J_2},$$

$$R(q_{1r}, \dot{q}_{1r}) = h\left(\dot{q}_{1r}^2 + H(q_{1r})\right)\sin(q_{1r}) - \frac{hK}{J_1}\dot{q}_{1r}\cos(q_{1r}) \tag{8}$$

Hence, we can take

$$\tau_r = H^{-1}(q_{1r})\left(u_r - a_0\eta - a_1\dot{\eta} - a_2\ddot{\eta} - R(q_{1r}, \dot{q}_{1r})\right) \tag{9}$$

where $H(q_r)$ is nonsingular around the equilibrium point $(q_{1r}^{\star}, \dot{q}_{1r}^{\star}) = (\pi, 0)$, a_0, a_1, and a_2 are positive constants. Introducing the new state coordinates $x = (x_1, x_2, x_3)^T = (\eta, \dot{\eta}, \ddot{\eta})^T$, we obtain

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix} = \underbrace{\begin{bmatrix}0 & 1 & 0\\ 0 & 0 & 1\\ -a_0 & -a_1 & -a_2\end{bmatrix}}_{A}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix} + \underbrace{\begin{bmatrix}0\\ 0\\ 1\end{bmatrix}}_{B}u_r, \tag{10}$$

$$\dot{p}_1 = -\frac{K}{J_1}p_1 + \frac{1}{J_1}y_r \tag{11}$$

$$y_r = \underbrace{\begin{bmatrix}1 & 0 & 0\end{bmatrix}}_{C}x. \tag{12}$$

Equation (11) constitutes the zero dynamics which is orbitally asymptotically stable [14]. The following *two-relay controller* is proposed for the purpose of exciting a periodic motion in (10):

$$u_r = -c_1\mathrm{sign}(y_r) - c_2\mathrm{sign}(\dot{y}_r) \tag{13}$$

where $c_1$and c_2 are scalars parameters designed such that the scalar-valued function output $y_r(t)$ has a periodic motion with the desired frequency Ω and amplitude A_p.

The formulas to compute c_1 and c_2, derived from the describing function method, are [6]

$$c_1 = \begin{cases} \dfrac{\pi}{4}\dfrac{A_p}{|W(j\Omega)|}\left(\sqrt{1+\xi^2}\right)^{-1} & \text{if } \Omega \in Q_2 \cup Q_3 \\ -\dfrac{\pi}{4}\dfrac{A_p}{|W(j\Omega)|}\left(\sqrt{1+\xi^2}\right)^{-1} & \text{elsewhere} \end{cases} \tag{14}$$

$$c_2 = \xi c_1 \tag{15}$$

where $W(j\omega) = C(j\omega I - A)^{-1}B$ is the corresponding transfer function of (10) and (12), $\xi = -\mathrm{Im}\{W(j\Omega)\}/\mathrm{Re}\{W(j\Omega)\}$, and the quadrants Q_i $(i = 1, \ldots, 4)$ are defined as

$$\begin{aligned} Q_1 &= \{\omega \in \mathbb{R}: \mathrm{Re}\{W(j\omega)\} > 0, \mathrm{Im}\{W(j\omega)\} \geq 0\} \\ Q_2 &= \{\omega \in \mathbb{R}: \mathrm{Re}\{W(j\omega)\} \leq 0, \mathrm{Im}\{W(j\omega)\} \geq 0\} \\ Q_3 &= \{\omega \in \mathbb{R}: \mathrm{Re}\{W(j\omega)\} \leq 0, \mathrm{Im}\{W(j\omega)\} < 0\} \\ Q_4 &= \{\omega \in \mathbb{R}: \mathrm{Re}\{W(j\omega)\} > 0, \mathrm{Im}\{W(j\omega)\} < 0\}. \end{aligned}$$

IV. CONTROL SYNTHESIS

The control objective is to stabilize the pendulum around the desired trajectory and to keep the wheel velocity bounded. This last can be achieved by introducing a feedback from the rotor velocity. In this Section, we will design a sliding modes tracking controller which ensures (4) while providing boundedness of $v_2(t)$ and attenuating external disturbances where the reference signal $q_r(t) \in \mathbb{R}^2$ is computed on line from (5), (9), and (13).

Let the pendulum tracking error and the wheel velocity error be defined respectively as

$$e_1(t) \triangleq q_{1r}(t) - q_1(t), \quad (18)$$
$$\dot{e}_1(t) \triangleq \dot{q}_{1r}(t) - v_1(t), \quad (19)$$
$$\dot{e}_2(t) \triangleq \dot{q}_{2r}(t) - v_2(t). \quad (20)$$

Due to (1), (5)-(13), the errors dynamics are governed by

$$\ddot{e}_1 = \ddot{q}_{1r} + \frac{J_2}{\Delta}\left(h\sin(q_{1r} - e_1) - f_s(\dot{q}_{2r} - \dot{e}_2)\right) - \frac{J_2}{\Delta}(w_1 - w_2) + \frac{J_2}{\Delta}\tau \quad (21)$$
$$\ddot{e}_2 = \ddot{q}_{2r} - \frac{J_2}{\Delta}h\sin(q_{1r} - e_1) + \frac{J_1}{\Delta}f_s(\dot{q}_{2r} - \dot{e}_2) - \frac{1}{\Delta}(J_1 w_2 - J_2 w_1) - \frac{J_1}{\Delta}\tau \quad (22)$$

where $\Delta = (J_1 - J_2)J_2$.

In order to achieve the control objective, let us propose the following sliding surface

$$s = \dot{e}_1 + m e_1 \quad (23)$$

where *m* is a positive constant. The following control law

$$\tau = -\frac{\Delta}{J_2}(M\text{sign}(s) + Ks - \gamma\text{sign}(\dot{e}_2) + m\dot{e}_1) - h\sin(q_{1r} - e_1) + f_s(\dot{q}_{2r} - \dot{e}_2) - \frac{\Delta}{J_2}\ddot{q}_{1r} \quad (24)$$

with parameters such that

$$M > 2WJ_2\Delta^{-1} + \gamma, \qquad K, \gamma > 0, \quad (25)$$

drives the trajectories to the sliding manifold $S = \{(q, \dot{q}) : s(e, \dot{e}) = \dot{s}(e, \dot{e}) = 0\}$. Using the fact that $J_1\ddot{q}_{1r} + J_2\ddot{q}_{2r} = -h\sin(q_{1r})$, the errors dynamics (21)-(22) result in

$$\ddot{e}_1 = -M\text{sign}(s) - Ks - m\dot{e}_1 + \gamma\text{sign}(e_2) - \frac{J_2}{\Delta}(w_1 - w_2) \quad (26)$$
$$\ddot{e}_2 = \frac{h}{J_2}\sin(q_{1r} - e_1) - \frac{h}{J_2}\sin(q_{1r}) + \frac{J_1}{J_2}\left(M\text{sign}(s) + Ks + m\dot{e}_1 - \gamma\text{sign}(\dot{e}_2)\right) - \frac{1}{\Delta}(J_1 w_2 - J_2 w_1). \quad (27)$$

Throughout, solutions of above system are defined in the sense of Filippov [15] as that of a certain differential inclusion with a multivalued right-hand side. The discontinuous controller (24) consists of the linear gain $-Ks$, the discontinuous part $-M\text{sign}(s)$ which ensure robustness and finite-time stability, and the rest are the nonlinear compensation terms.

Theorem 1. *The controller introduced in (23)-(24) subject to parameters (25) ensures asymptotic stability of the equilibrium point* $\varphi^e = (e_1^e, \dot{e}_1^e, \dot{e}_2^e)^T = 0$.

Proof. We break the proof in two steps.
1. Convergence of $(e_1, \dot{e}_1)$ *to the origin*. First, we will demonstrate the existence of sliding modes by verifying $s\dot{s} < 0$ [16]. Therefore

$$s\dot{s} = s[\ddot{e}_1 + m\dot{e}_1]$$

Substituting (21), (23), and (24) into above equation we have

$$s\dot{s} < -\left[M - \gamma - 2\frac{J_2}{\Delta}\right]|s| - Ks^2 < 0$$

where inequalities, due to (3),

$$\frac{J_2}{\Delta}(w_1 - w_2) \leq \frac{J_2}{\Delta} \parallel w_1 - w_2 \parallel \leq 2\frac{J_2}{\Delta}W \tag{29}$$

$$\frac{1}{\Delta}(J_1 w_2 - J_2 w_1) \leq \frac{1}{\Delta} \parallel J_1 w_2 - J_2 w_1 \parallel \leq 2\frac{J_1}{\Delta}W \tag{30}$$

has been used. Since $s\dot{s}$ is negative definite it is then concluded that the trajectories reach the sliding manifold *S* asymptotically. Moreover, to conclude that trajectories reach *S* in finite-time we take $V_1 = s^2/2$, thus

$$\begin{aligned}\dot{V} = s\dot{s} &< -\left[M - \gamma - 2\frac{J_2}{\Delta}\right]|s| \\ &= -2\left[M - \gamma - 2\frac{J_2}{\Delta}\right]\sqrt{V}\end{aligned}$$

which implies

$$\frac{dV}{\sqrt{V}} < -2\left[M - \gamma - 2\frac{J_2}{\Delta}\right]t.$$

Integrating the above inequality from 0 to t_f we conclude that

$$t_f < \left[M - \gamma - 2\frac{J_2}{\Delta}\right]^{-1}\sqrt{V(0)}.$$

Then it is concluded that trajectories of (26) is reached in finite-time t_f. It is also concluded from (23) that, once the trajectories reach the sliding manifold *S*, then $e_1(t)$ and $\dot{e}_1(t)$ tend to the origin asymptotically with a decay rate *m*.

Now, we will proof that $\dot{e}_2$ will tends to the origin from t_f to infinity. To this end, note that the remaining dynamics of the velocity error of the wheel (27) is reduced to

$$\ddot{e}_2 = -J_1 J_2^{-1}\gamma \text{sign}(\dot{e}_2).$$

Using the Lyapunov function $\dot{V}_2(\dot{e}_2) = J_1^{-1}J_2|\dot{e}_2|$ we have that

$$\dot{V}_2(\dot{e}_2) = -\gamma$$

therefore $\dot{V}_2(\dot{e}_2)$ will be negative definite for all $t \geq t_f$.

2. Boundedness of wheel velocity. Finally, we need to prove that $\dot{e}_2$ does not escape to infinity in finite time, that is, $\lim_{t \to t_f}\|\dot{e}_2(t)\| < \infty$. To this end, let us rewrite (27) as

$$\ddot{e}_2 = -J_1 J_2^{-1}\gamma \text{sign}(\dot{e}_2) + g(e_1, \dot{e}_1, w) \tag{31}$$

where

$$g(e_1, \dot{e}_1, w) = \frac{h}{J_2}\sin(q_{1r} - e_1) - \frac{h}{J_2}\sin(q_{1r}) - \frac{1}{\Delta}(J_1 w_2 - J_2 w_1) + \frac{J_1}{J_2}(M\text{sign}(s) + Ks + m\dot{e}_1). \tag{32}$$

Since (29), (30) and since e_1, $\dot{e}_1$, and *s* decreases asymptotically to the origin (that is, $|e_1(t)| \leq |e_1(0)|$ and $|\dot{e}_1(t)| \leq |\dot{e}_1(0)|$), the following upper bound for $g(e_1, \dot{e}_1, w)$ is obtained

$$\|g(e_1, \dot{e}_1, w)\| \leq \frac{2h}{J_2} + 2\frac{J_1}{\Delta}W + \frac{J_1}{J_2}(M + K|s(0)| + m\dot{e}_1(0)) = \varepsilon_0. \tag{33}$$

Let us consider now the following Lyapunov function

$$V(\dot{e}_2) = J_1^{-1}J_2|\dot{e}_2|. \tag{34}$$

The time derivative of $V(\dot{e_2})$ along the solution of the closed-loop system (31)-(32) is

$$\begin{aligned}\dot{V}(\dot{e_2}) &= -J_1^{-1}J_2\mathrm{sign}(\dot{e}_2)\ddot{e}_2\\ &= -J_1^{-1}J_2\mathrm{sign}(\dot{e}_2)(-J_1J_2^{-1}\gamma\mathrm{sign}(\dot{e}_2)+g(e_1,\dot{e}_1,w))\\ &= -\gamma - J_1^{-1}J_2\mathrm{sign}(\dot{e}_2)g(e_1,\dot{e}_2,w)\\ &\le -\gamma + J_1^{-1}J_2\|g(e_1,\dot{e}_1,w)\|\\ &\le -\gamma + J_1^{-1}J_2\varepsilon_0\end{aligned} \tag{35}$$

where inequality (33) has been used. Due to inequality (25), the time derivative of the Lyapunov function $\dot{V}(\dot{e_2})$ is not negative definite. However, integrating $\dot{V}(\dot{e_2}(t))$ from 0 to t_f, we get

$$V(t_s) \le (J_1^{-1}J_2\varepsilon_0 - \gamma)t_f + V(0) \tag{36}$$

which implies

$$|\dot{e}_2(t_s)| \le (\varepsilon_0 - J_1J_2^{-1}\gamma)t_f + |\dot{e}_2(0)|. \tag{37}$$

Therefore, we conclude that $\dot{e}_2$ is finite for $t_f < \infty$. The proof is completed.

V. Simulation Results

In this Section, we present numerical results using the laboratory inertial wheel pendulum manufactured by Quanser Inc., depicted in Fig. 1 where $J_1 = 4.572 \times 10^{-3}$ Kg-m^2, $J_2 = 2.495 \times 10^{-5}$ Kg-m^2, and h = 0.3544 Kg-m^2 s^{-2} (see [17]). The viscous friction coefficient $f_s = 8.80 \times 10^{-5}$ Nm-s was identified by applying the procedure from [18]. Simulations were carried out to achieve the orbital stabilization of the unactuated link (pendulum) q_1 around the equilibrium point $q^\star = (\pi, 0)$.

Setting $\Omega = \pi/2$ rad/s and $A_p = 0.07$ as desired frequency and amplitude, respectively, we have $c_1 = 2$ and $c_2 = -2$ which have been computed from (14) and (15). Invoking [6, Thm. 2.1], the system (10) was orbital asymptotically stable under K = 1, $a_0 = 1$, $a_1 = 2$, and $a_2 = 1$.

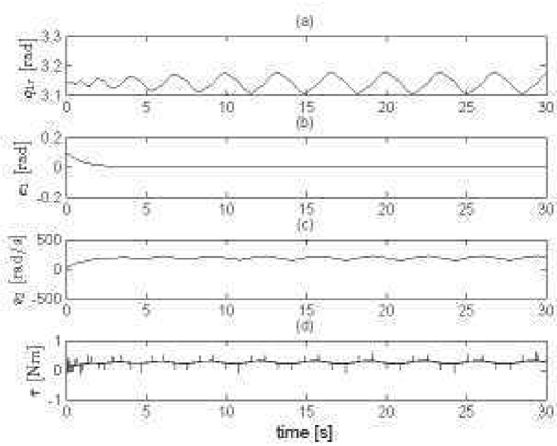


Fig. 1: Simulation results for the unperturbed case: (a) reference trajectory, (b) link position error, (c) wheel velocity, and (d) control input.

The initial conditions of the inertia wheel pendulum, selected for the simulations, were close to $q_1(0) = 3$ rad and $q_2(0)$ =0 rad, whereas all the velocity initial conditions were set to $v_1(0) = v_2 = 0$ rad/s. The gains of the controller (23) were set to M = 2, m = 1, and $\gamma = 0.01$.

Figure 2 shows the desired trajectory to follow, the tracking pendulum position error, the wheel velocity, and the input torque for the disturbance-free case. This Figure corroborates that sliding mode controller guarantee finite-time stabilization of the pendulum around the desired trajectory. In order to test the controller robustness, harmonic external disturbance $w_i = 0.005\sin(t)$ $(i = 1{,}2)$ is then applied to the closed-loop system. Good performance and desired robustness properties of the controller are concluded from Fig. 3.

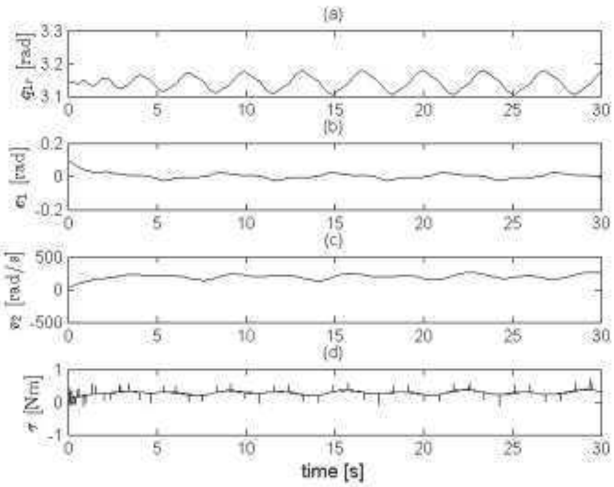


Fig. 2: Simulation results for the perturbed case: (a) reference trajectory, (b) link position error, (c) wheel velocity, and (d) control input.

VI. CONCLUSIONS

In this paper we have synthesized a sliding mode controller to solve the tracking control problem of the inertia wheel pendulum. The periodic motion of the pendulum will be at the upright position around the unstable equilibrium point of the unforced system where the two-relay controller was also used to induce of oscillations at the scalar output of the inertia wheel pendulum reference model and then injected as desired trajectory to the closed-loop system. Such configuration allows to generate oscillations without depend of the neglected dynamics. Therefore, the resulting control law ensures rejection against bounded matched disturbances while numerical study verifies the performance of the synthesized controller.

REFERENCES

[1] Chevallereau, C., Abba, G., Aoustin, Y., Plestan, E., Canudas-de-Wit, C., and Grizzle, J. "Rabbit: a testbed for advanced control theory," *IEEE Control Systems Magazine*, vol. 23, no. 5, pp. 57–79, 2003.

[2] Hurmuzlu, Y., Génot, F., and Brogliato, B. "Modeling, stability, and control of biped robots—a general framework," *Automatica*, vol. 40, pp. 1647–1664, 2004.

[3] Plestan, F., Grizzle, J.,Westervelt, E., and Abba, G. "Stable walking of a 7-DOF biped robot," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 4, pp. 653–668, 2003.

[4] Albea, C., Canudas-de-Wit, C., and Gordillo, F. (2007). "Adaptive control of the boost DC-AC converter," *Proceedings of the IEEE International Conference on Control Applications*, 611–616.

[5] Olivier, J.C., Le Claire, J.C., and Loron, L. "An efficient switching frequency limitation process applied to high dynamic voltage supply," *IEEE Transactions on Power Electronics*, vol. 23, no. 1, 153–162, 2008.

[6] Aguilar, L., Boiko, I., Fridman, L., and Iriarte, R. *Self-Oscillations in Dynamic Systems—A New Methodology via Two-Relay Controllers*. Birkhauser, London, UK, 2015.

[7] Aguilar, L., Boiko, I., Fridman, L., and Freidovich, L. "Generating oscillations in inertia wheel pendulum via two-relay controller," *International Journal of Robust and Nonlinear Control*, vol. 22, pp. 318–330, 2012.

[8] Aguilar, L., Boiko, I., Fridman, L., and Iriarte, R. Generating self-excited oscillations for underactuated mechanical systems via two-relay controller. *International Journal of Control*, vol. 82, no. 9, 1678–169, 2009.

[9] Aguilar, L., Boiko, I., Fridman, L., and Iriarte, R. "Generating self-excited oscillations via two-relay controller," *IEEE Transactions on Automatic Control*, vol. 54, no. 2, 416–420, 2009.

[10] Khalil, H. *Nonlinear Systems*. Third edition, Prentice Hall, Upper Saddle River, 2002.

[11] Block, D., Astrom, K., and Spong, M. *The Reaction Wheel Pendulum*. Synthesis Lectures on Control and Mechatronics #1. Morgan & Claypool Publisher, 2007.
[12] Iriarte, R., Aguilar, L., and Fridman, L. "Second order sliding mode tracking controller for inertia wheel pendulum," *Journal of the Franklin Institute*, vol. 350, pp. 92–106, 2013.
[13] Grizzle, J., Moog, C., and Chevallereau, C. "Nonlinear control of mechanical systems with an unactuated cyclic variable," *IEEE Transactions on Automatic Control*, vol. 50, no. 5, pp. 559–576, 2005.
[14] Wiggins, S. *Introduction to Applied Nonlinear Dynamical Systems and Chaos*. Text in Applied Mathematics, 2nd edn. Springer, New York, 2000.
[15] Filippov, A. *Differential Equations with Discontinuous Right-hand Sides*. Kluwer Academic Publisher, Dordrecht, 1988.
[16] Utkin, V. *Sliding Modes in Control Optimization*. Springer-Verlag, Berlin, 1992.
[17] Astrom, K., Block, D., and Spong, M. *The Reaction Wheel Pendulum*. Lecture Notes for the Reaction Wheel Pendulum (Part of the Mechatronics Control Kit). Urbana-Champaigne, IL, 2001.
[18] Kelly, R., Llamas, J., and Campa, R. "A measurement procedure for viscous and Coulomb friction," *IEEE Transactions on Instrumentation and Measurements*, vol. 49, no. 4, pp. 857–861, 2000.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 7, páginas 59 - 69.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

On adaptive control of a permanent magnet synchronous motor[1]

J. Moreno–Valenzuela[2], Y. Quevedo-Pillado, R. Pérez-Aboytes and L. González–Hernández [3]

***Abstract*—This paper is inspired on the structure of the field oriented control by presenting a practical study of an adaptive nested controller for trajectory tracking control of a permanent magnet synchronous motor (PMSM) driving a single–link arm. It is worthwhile to notice that the new controller resembles the pure field oriented control except by the adaptive terms. Real–time experimental implementations have been carried out in order to asses the performance of the new controller with respect to the classical non adaptive field oriented control algorithm.**

I. Introduction

Electric motors are the most widely prime movers used in all over the world in industrial and services applications (e.g., machine tools, compressors, fans, robots, etc.) and home appliances [1]. Specifically, special attention has been devoted to the permanent magnet synchronous motor (PMSM) due to its capability of a greater torque with respect to other electric motors [2]. But, the control of the PMSM is difficult due to its multi input nature and highly coupled by far more nonlinear dynamics, which makes it a topic that has attracted the attention of the control community since the eighties until now.

The most notable technique to improve the performance of the PMSM is based on field oriented control first proposed in [3] and further detailed in [4]. Field oriented control allowed controllers to be developed by attaching a rotating reference frame to the rotor via the Park transform. Once posed in this reference frame, trigonometric nonlinearities are removed, resulting in a bilinear system that allows for independent regulation of currents. Field oriented control is the preferred scheme in industrial applications, due to its structure based on nested proportional-integral loops [5].

It is of interest to mention that there is an ample literature revealing that many theorists and practitioners have developed control schemes for the PMSM using different type of approaches. See, e.g., [6], [7], [8], [9], [10], [11], [12], for example.

While many controllers have devised to solve position and velocity regulation, the amount of works exploring the trajectory tracking control of the PMSM driving a robotic load [13], [14], [15], seems to be still limited.

This document presents a new adaptive motion control of PMSM driving a single–link arm. The complexity in this system is increased due to the nonlinearly introduced by effect of the gravity in the arm. The new scheme is based in the adaptation of 4 electrical parameters and 3 mechanical parameters. The results in this document are supported by real–time experimental results.

The organization of the paper is as follows: Section 2 presents the model of the system and the problem formulation. In Section 3, the new scheme is described. Real–time experimental results are described in Section 4. Finally, the conclusions derived from our study are given in Section 5.

II. System dynamics and control problem

A. System dynamics

The system considered in this document is compound by a PMSM and an arm, which is also denoted

[1] This work was supported by Secretara de Investigación y Posgrado–IPN, CONACyT, and DGEST, Mexico.
[2] Corresponding author. Email: moreno@citedi.mx, emorenov@ipn.mx.
[3] All authors are with Instituto Politécnico Nacional-CITEDI, Instituto Politécnico Nacional No. 1310, Col. Nueva Tijuana, Tijuana, B.C., México, 22435, fax: +52 (664) 623 1388.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

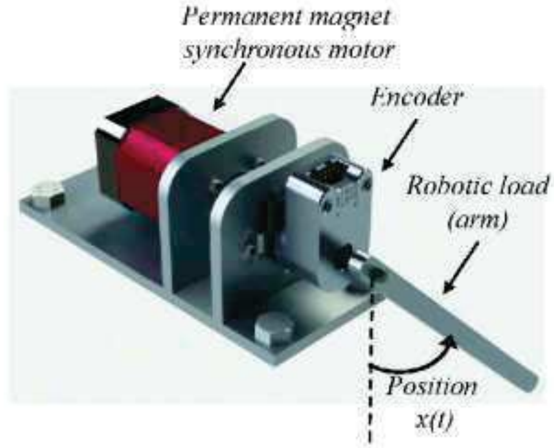


Figure 1: CAD drawing of the PMSM driving a robotic load.

as the robotic load; see Figure 1. The mathematical model of the electrical part is given by [14], [1], [11],

$$L_d \frac{d}{dt} I_d = -RI_d - n_p L_q I_q \dot{x} - K_\tau \dot{x} + V_d, \quad (1)$$

$$L_q \frac{d}{dt} I_q = -RI_q + n_p L_d I_d \dot{x} + V_q, \quad (2)$$

where $I_i(t)$, $V_i(t)$ and L_i (with $i = d, q$), are current, voltage and inductance based on the (d, q) quadrature coordinates, R is a positive constant denoting the winding resistance, n_p is the number of pole pairs of permanent magnets in the rotor, K_τ is a positive constant, and $\dot{x}(t)$ is the rotor angular velocity.

The equation for the mechanical dynamics is

$$M \frac{d}{dt} \dot{x} = -B\dot{x} - N \sin(x) + [K_q I_q + 1] I_d, \quad (3)$$

were M denotes the lumped mechanical inertia, N is the load lumped constant, B is the lumped viscous friction coefficient, K_q is a positive constants for torque transmission, and $x(t)$ is the rotor position.

B. Control problem

Let us assume that the states $x(t)$, $\dot{x}(t)$, $I_d(t)$ and $I_q(t)$ are available through proper sensors in the system. The control problem is to design control inputs V_d and V_q so that

$$\lim_{t\to\infty} \tilde{x}(t) = 0, \quad (4)$$

where

$$\tilde{x}(t) = x_d(t) - x(t) \quad (5)$$

is the position tracking error and $x_d(t)$ represents the desired position trajectory. The desired trajectory $x_d(t)$ is assumed to be continuous and three times differentiable. We also assumed that the signals $x_d(t)$, $\dot{x}_d(t)$, $\ddot{x}_d(t)$ and $x_d^{(3)}(t)$ are bounded for all time $t \geq 0$.

Adaptive control considers the use of a dynamic extension in order to estimate on–line some constant parameters related to the system such as the inertia, viscous friction and gravitational load. Adaptive control is a direct aggregation of a control methodology with some form of recursive system identification [16].

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

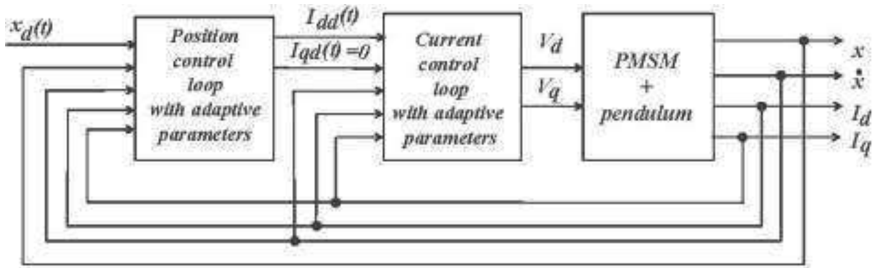


Figure 2: Block diagram of the nested current/position control scheme. Adaptive parameters are considered in each one of the two control loops.

III. A NEW ADAPTIVE CONTROLLER

The nested structure of current/position control for velocity regulation was just recently proven to be asymptotically stable in [12]. As pointed out there, there is still theoretical and practical work to understand the efficiency and advantages of using a nested control structure with the current control loop as the core of the system. With this aim, here we propose an adaptive controller which is based in the adaptation of 7 parameters and consists in two nested loops:

• Adaptive current control loop, which has the goal of achieving tracking of the desired current commands provided by the position loop.

• Adaptive position control loop, which deals with desired current commands in order to guarantee tracking of a time–varying position reference. This control loop also considers the presence of a robotic load.

See Figure 2 for a general block diagram of the nested current/position controller with parameter adaption. The novelty relies in considering parameter adaption as well as providing a rigorous stability analysis, which is given in Section 4.

A. Adaptive control of the electrical part

The current control loop is designed with the aim of ensuring tracking of the desired current commands. In particular, the desired current for the phase d, denoted as $I_{dd}(t)$, is assumed to be differentiable and the desired current for the phase q is defined null, that is,

$$I_{qd}(t) = 0, \quad \forall\, t \geq 0. \tag{6}$$

In the next, phased voltages $V_d(t)$ and $V_q(t)$ are designed to guarantee that the current errors asymptotically converge to zero.

In order to facilitate the analysis, it will be convenient rewrite the electrical dynamics (1) in form of model regressor as

$$W_{ed}\theta_{ed} = V_d, \tag{7}$$

where

$$W_{ed} = \begin{bmatrix} \dot{I}_d & I_d & n_p I_q \dot{x} & \dot{x} \end{bmatrix} \in \mathbb{R}^{1\times 3}, \tag{8}$$

and

$$\theta_{ed} = \begin{bmatrix} L_d & R & L_q & K_\tau \end{bmatrix} \in \mathbb{R}^{4}, \tag{9}$$

It is possible to use the vector θ_{ed} in (9) with a proper definition of a regression matrix W_{eq} to represent the electrical dynamics (2) so that

$$W_{eq}\theta_{ed} = V_q, \tag{10}$$

where

$$W_{eq} = \begin{bmatrix} -n_p I_d \dot{x} & I_q & \dot{I}_q & 0 \end{bmatrix} \in \mathbb{R}^{1\times 4}, \tag{11}$$

where W_{ed} and W_{eq} are defined as regressor matrices while θ_{ed} is a constant parameter vector.

In order to introduce the control inputs $V_d(t)$ and $V_q(t)$, it will be convenient to define the current errors

$$\eta_d(t) = I_{dd}(t) - I_d(t), \tag{12}$$

$$\eta_q(t) = I_{qd}(t) - I_q(t) = -I_q(t), \tag{13}$$

since $I_{qd}(t)$ was defined null in equation (6).

It is useful to define

$$\hat{\theta}_{ed}(t) = \begin{bmatrix} \hat{L}_d(t) \\ \hat{R}(t) \\ \hat{L}_q(t) \\ \hat{K}_{\tau 2}(t) \end{bmatrix} \mathbb{R}^4, \tag{14}$$

which denotes an estimation at time t of the vector of constant parameters for the phase d given by θ_{ed} in (9). The vector (14) is obtained by an update law to be defined later and is used in the control inputs $V_d(t)$ and $V_q(t)$. The proposed voltages to control the electrical dynamics in (1)-(3), which was rewritten in (7)–(11), are

$$V_d = W_{ed}^d \hat{\theta}_{ed} + K_{pd}\eta_d + K_{id}\xi_d, \tag{15}$$

$$\dot{\xi}_d = \eta_d, \tag{16}$$

for the phase d, and

$$V_q = W_{eq}^d \hat{\theta}_{ed} + K_{pq}\eta_q + K_{iq}\xi_q, \tag{17}$$

$$\dot{\xi}_q = \eta_q. \tag{18}$$

for the phase q. It is noteworthy to say that the controllers in (15)–(16) and (17)–(18) have a PI part in the current error plus an adaptive model regressor part. More specifically, $\eta_d(t)$ and $\eta_q(t)$ are defined in (12) and (13), respectively, the signals $I_{dd}(t)$, which is actually the output of the position control loop to be defined in the next, and $I_{qd}(t)$, defined null in (6), represent the desired current trajectories for each one of the motor phases, the desired regression matrix for phase d is given by

$$W_{ed}^d = \begin{bmatrix} \dot{I}_{dd} & I_{dd} & n_p I_q \dot{x} & \dot{x} \end{bmatrix} \in \mathbb{R}^{1\times 4}, \tag{19}$$

$$\hat{\theta}_{ed} = \begin{bmatrix} \hat{L}_d & \hat{R} & \hat{L}_q & \hat{K}_{\tau 2} \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^4 \tag{20}$$

is the vector of estimated parameters of the phase d, K_{pd} y K_{pq} are positive proportional gains, K_{id} and K_{iq} are positive integral gains, the desired regression matrix for phase q is defined by

$$W_{eq}^d = [-n_p I_d \dot{x}\ 0\ 0\ 0] \in \mathbb{R}^{1\times 4}. \tag{21}$$

the vector $\hat{\theta}_{ed}$ was defined in (20), and signals ξ_d, ξ_q are the integrals with respect to time of the current errors η_d, η_q, respectively.

The purpose of using the definition of the estimated parameters $\hat{\theta}_{ed}$ in the control inputs $V_d(t)$ and $V_q(t)$, in (15) and (17) is to simplify the notation and simplify the derivation of stability analysis.

Finally, it remains to introduce the adaption law to obtain the real–time update of the vector $\hat{\theta}_{ed}(t)$ used in the controller $V_d(t)$ in (15) and $V_q(t)$ in (17):

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones.   Capítulo 7, páginas 59 - 69.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

$$\frac{d}{dt}\hat{\theta}_{ed} = \Gamma_e[W_{ed}^{dT}\eta_d + W_{eq}^{dT}\eta_q], \tag{22}$$

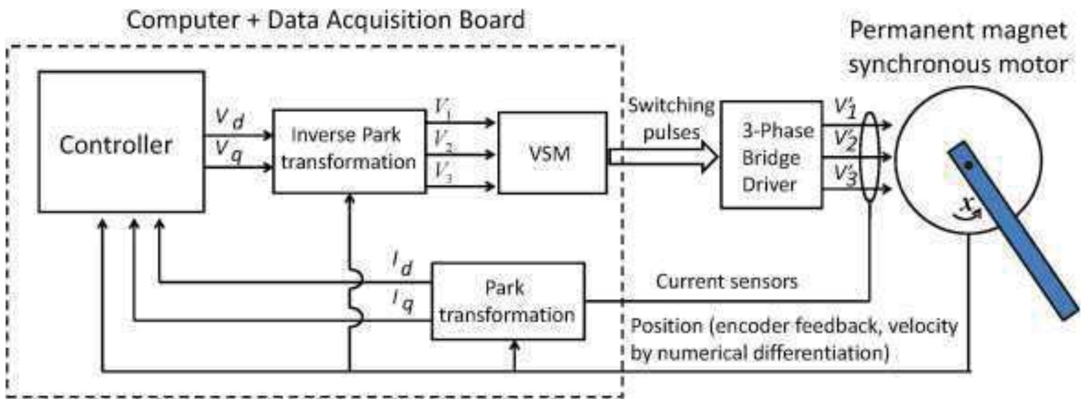


Figure 3: Block diagram of the experimental platform. Velocity $\dot{x}(t)$ is estimated via numerical differentiation. The block VSM means vector space modulation, which is achieved by software and then switching pulses are computed for the 3–phase bridge driver.

where Γ_e is 4×4 symmetric positive definite matrix and the desired regression matrices W_{ed}^d and W_{eq}^d are defined in (19) and (21), respectively. The signals η_d and η_q are the phase current errors in (12) and (13), respectively.

B. Adaptive control of the mechanical part

Here, the controller of the mechanical part of the system (3)–(2) is presented. By considering that I_{qd} is null, which was previously defined in (6), and replacing the definitions of the current errors in (12)–(13) into the mechanical dynamics (3), we obtain

$$\begin{aligned} M\ddot{x} + B\dot{x} + N\sin(x) &= [-K_q\eta_q + 1][I_{dd} - \eta_d] \\ &= I_{dd} + w(I_{dd}, \eta_d, \eta_q) \end{aligned} \tag{23}$$

where

$$w(I_{dd}, \eta_d, \eta_q) = K_q\eta_q\eta_d - K_q I_{dd}\eta_q - \eta_d \tag{24}$$

is perturbation introduced by the current errors η_d and η_q. A regression form can be used to write (23) as follows

$$W_\tau\theta_\tau = I_{dd} + w(I_{dd}, \eta_d, \eta_q),$$

where

$$W_\tau = \begin{bmatrix}\ddot{x} & \dot{x} & \sin(x)\end{bmatrix} \in \mathbb{R}^{1\times3}$$

is the regression matrix of the mechanical part and

$$\theta_\tau = \begin{bmatrix}M & B & N\end{bmatrix}^T \in \mathbb{R}^3$$

is the vector of constant parameters of the mechanical model assumed to be unknown. The signal w is defined in (24) and as previously pointed out denotes the perturbation introduced by the current errors.

Thus, we propose the following desired current for the phase d, which indirectly controls the mechanical part of the system:

$$\begin{aligned} I_{dd} &= \hat{M}\ddot{x}_d + \hat{B}\dot{x}_d + \hat{N}\sin(x_d) + K_p\tilde{x} + K_d\tilde{\vartheta} \\ &= W_\tau^d\hat{\theta}_\tau + K_p\tilde{x} + K_d\tilde{\vartheta}, \end{aligned} \tag{25}$$

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

where $\tilde{x} \in \mathbb{R}$ is the position tracking error which was defined in (5),

$$W_\tau^d = \begin{bmatrix} \ddot{x}_d & \dot{x}_d & \sin(x_d) \end{bmatrix} \mathbb{R}^{1\times 3}, \tag{26}$$

is the regression matrix of the mechanical part of the model,

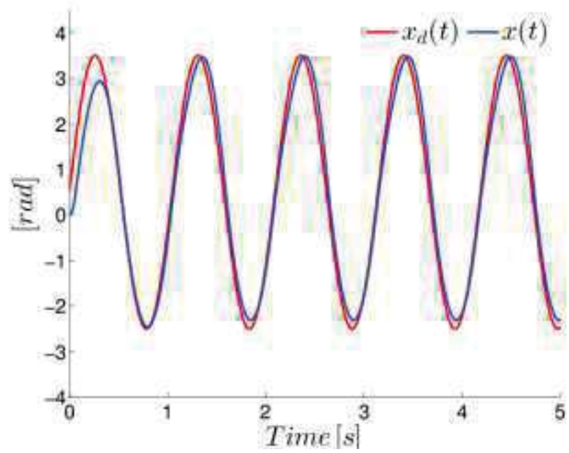


Figure 4: Classical non–adaptive field oriented control: Time evolution of the actual position $x(t)$.

$$\hat{\theta}_\tau = \begin{bmatrix} \hat{M} & \hat{B} & \hat{N} \end{bmatrix}^T \in \mathbb{R}^3, \tag{27}$$

is the vector of estimated mechanical parameters, and $\tilde{\vartheta}$ is obtained by

$$\dot{\tilde{\vartheta}} = -K_f[\tilde{\vartheta} - \dot{\tilde{x}}], \tag{28}$$

where $K_f > 0$.

Let us notice that the purpose of using the filtered velocity $\tilde{\vartheta}(t)$ from (28) is to avoid acceleration measurements $\ddot{\tilde{x}}$ in the calculation of the signal $\dot{I}_{dd}(t)$, which at the same time is used in the controller $V_d(t)$ in (15).

The desired current command $I_{dd}(t)$ in (25) is designed to drive the mechanical part of system (3) so that asymptotic vanishing of the position error $\tilde{x}(t)$ in (5) be guaranteed. As seen, I_{dd} in (25) has the structure of a PD controller plus adaptive feed forward compensation of mechanical part.

In order to update the signal $\hat{\theta}_\tau(t)$, the adaptation law is proposed

$$\frac{d}{dt}\hat{\theta}_\tau = \Gamma_\tau W_\tau^{dT}[\dot{\tilde{x}} + \alpha\tilde{x}], \tag{29}$$

with Γ_τ a 3×3 positive definite matrix, W_τ^d as in (26), and α is strictly positive constant whose selection criteria is given in Proposition 1, stated in the next Section.

IV. Experimental Resuls

A. Description of the experimental platform

The experimental tests have been conducted in a PMSM motor prototype built at *Instituto Politécnico Nacional-CITEDI*. The hardware components used for the instrumentation and control of the PMSM motor experimental platform are as follows:

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

- desktop personal computer (PC),
- PCI data acquisition (DAQ) board Sensoray 626,

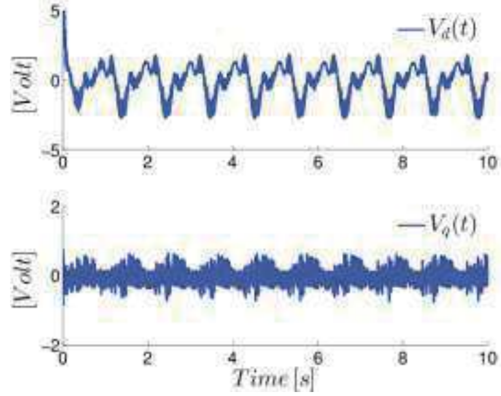


Figure 5: Classical non–adaptive field oriented control: Time evolution of the applied voltage $V_d(t)$ and $V_q(t)$.

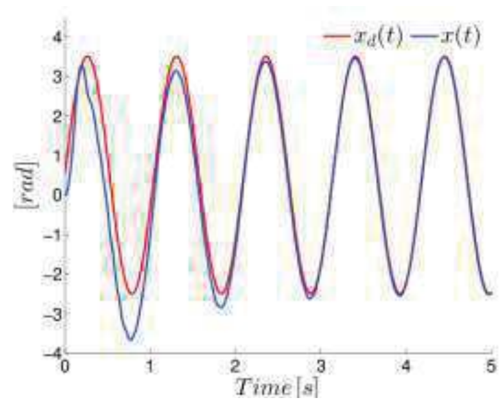


Figure 6: New adaptive controller: Time evolution of the actual position $x(t)$.

- current sensors NT–5 from FW Bell,
- motor Anaheim Automation BLY172S-24V-4000 with $n_p = 4$,
- optical encoder HB5M-1000-250-IE-D-H from US Digital, and
- a 3–phase bridge driver based in the IR2130 circuit from International Rectifier.

The PC runs Windows XP operative system, Matlab and Simulink, which interacts with the DAQ through the Real-Time Windows Target libraries. The programming of the control algorithms is based in the computation of the d - q voltage V_d and V_q, the inverse Park transformations to get the three phase voltages, and the vector space modulation, which computes the required switching for the transistors in the 3–phase bridge driver. The controller is updated at 1 kHz while the the vector space modulation runs at 10 kHz. See also Figure 3 for a block diagram implementation of the experimental platform.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

B. Experimental results using the classical non–adaptive field oriented control (CNAFOC)

If the adaptive part of the proposed controller is removed, the classical non–adaptive field oriented control (CNAFOC) is obtained [5], [11], [12],

$$V_d = K_{pd}\eta_d + K_{id}\xi_d, \tag{30}$$

$$V_q = K_{pq}\eta_q + K_{iq}\xi_q, \tag{31}$$

where the current errors $\eta_d(t)$ and $\eta_q(t)$ are defined in (12) and (13), respectively, the signals $\xi_d(t)$ and $\xi_q(t)$ are obtained from (16) and (18), respectively, and the desired currents are

$$I_{dd} = K_p e + K_d \dot{e}, \tag{32}$$

$$I_{qd} = 0, \tag{33}$$

where the velocity error signal $\dot{e}(t)$ is used instead of $\tilde{\vartheta}(t)$ obtained from filter (28).

The CNAFOC has been implemented in our experimental platform. The desired trajectory used in this test is

$$x_d(t) = 3\sin(6t) + 0.5\,[rad]. \tag{34}$$

The purpose of using this trajectory is to maximize the effect of the gravity in the pendular load and then to assess the tracking performance.

The gains used for this experiment are

$$\begin{array}{cc} K_p = 2.5, & K_d = 0.75, K_{pd} = 0.75, \\ K_{id} = 0.75, K_{pq} = 0.75, & K_{iq} = 0.75. \end{array} \tag{35}$$

The results are observed in Figure 4, which shows the time evolution of the desired trajectory $x_d(t)$ and the actual one $x(t)$, and in Figure 5, which depicts the applied voltage $V_d(t)$ and $V_q(t)$.

C. Experimental results using the controller

The new adaptive controller given in equations in (15)–(17), has been implemented using the control gains (35), filter gain $K_f = 1000$.

Besides, the matrix Γ_e used in the electrical parameter adaptation law in (22) was taken as diagonal matrix with the following gains

$$\Gamma_{e1} = 2.4\times10^{-6}, \Gamma_{e2} = \Gamma_{e3} = \Gamma_{e4} = 0.24\times10^{-3},$$

Matrix Γ_τ using in the mechanical parameters adaptation law (29) was taken taken also diagonal with the following gains

$$\Gamma_{\tau1} = 0.01, \Gamma_{\tau2} = 0.6, \Gamma_{\tau3} = 6.0, \alpha = 1.0.$$

The results for the new adaptive controller are shown as follows: Figure 6 shows the time evolution of the actual position trajectory $x(t)$ and the desired position $x_d(t)$, Figure 7 depicts the applied phase voltage $V_d(t)$ and $V_q(t)$, the estimated electrical parameters $\hat{L}_d(t)$, $\hat{L}_q(t)$, $\hat{R}(t)$ and $\hat{K}_{\tau2}(t)$ are observed in Figure 9, and finally the result for the estimated mechanical parameters $\hat{M}(t)$, $\hat{B}(t)$ and $\hat{N}(t)$ are illustrated in Figure 8.

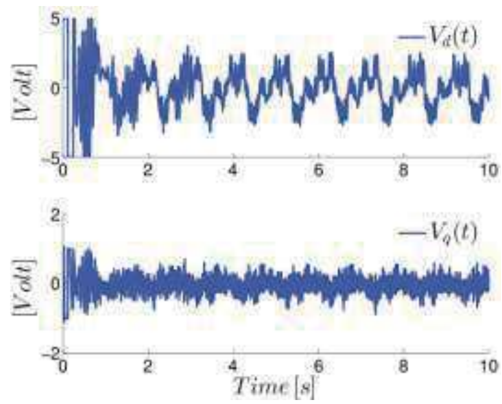


Figure 7: New adaptive controller: Time evolution of the applied voltage $V_d(t)$ and $V_q(t)$.

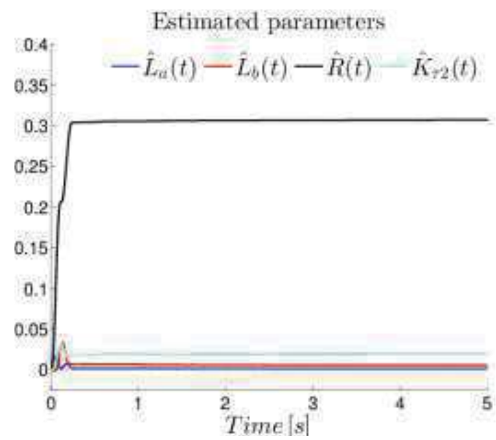


Figure 8: New adaptive controller: Time evolution of the estimated electrical parameters.

D. Observations

We have computed the root mean square (RMS) values of the position error $e(t)$, the current error vector $[\eta_d(t)\,\eta_q(t)]^T$ and the control input vector $[V_d(t)\,V_q(t)]^T$ in the time interval $5 \leq t \leq 10$ $[s]$. We have considered that for all time $t \geq 5$ transients are vanished. The results are given in Table 1.

Notice that the RMS value of the tracking error $e(t)$ obtained for the new adaptive controller is improved 81% with respect to the CNAFOC.

Besides, the RMS value of the applied voltage vector $[V_d(t)V_q(t)]^T$ in only 7.8% bigger than the obtained for the classical non–adaptive field oriented control scheme. In other words, a small difference in the applied energy implies a significant improvement in the tracking error $e(t)$.

Visual examination of Figures 4 and 6 reveals that actual trajectory $q(t)$ for implementation of the new controller (15)–(17) follows with more accuracy the desired trajectory $x_d(t)$.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

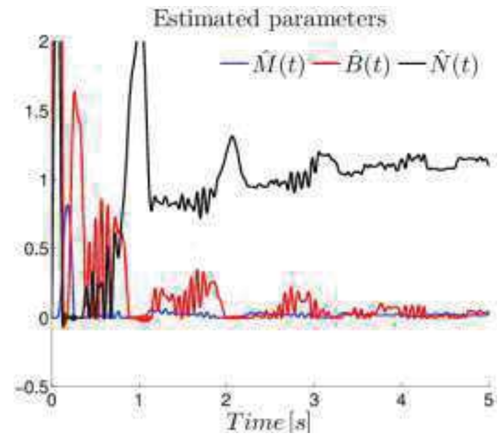


Figure 9: New adaptive controller: Time evolution of the estimated mechanical parameters.

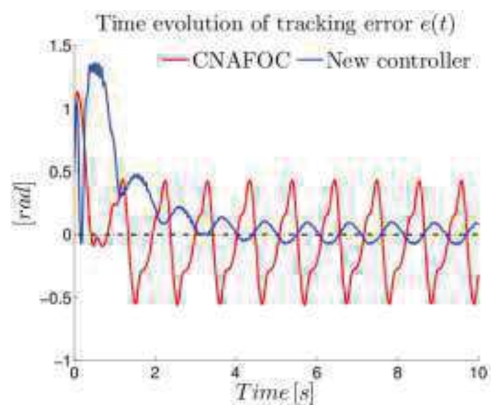


Figure 10: Comparison of the time evolution of the tracking error $e(t)$ for both controllers. CNAFOC means classical non–adaptive field oriented control.

Table 1: Performance of the classical non–adaptive field oriented control (CNAFOC) and new adaptive controller.

Index	CNAFOC	New	Unit
$RMS(e)$	0.3046	0.0579	$[rad]$
$RMS(\eta_d, \eta_q)$	1.440	0.824	$[Amp]$
$RMS(V_d, V_q)$	1.076	1.160	$[Volt]$
$\max_{t \geq 5}(PTPV)$	0.98	0.17	$[rad]$

In order to have a better idea of the obtained performance, the time evolution of the tracking error $e(t)$ for both controllers, is given in Figure 10. For $5 \leq t \leq 10$, we have computed the maximum peak–to–peak value (PTPV) of the tracking error $e(t)$ for both controllers, which is shown in Table 1. The

maximum PTPV obtained for new adaptive controller (15)–(17) is much smaller than the one obtained for the classical non–adaptive field oriented control (30)–(31). Specifically, the improvement is of 82.3%.

V. CONCLUSIONS

In this document, a new adaptive trajectory tracking controller for the PMSM driving a robotic load was introduced. The new scheme was inspired in the field oriented control approach, that is, in the use of Park's transformation and nested current/position loops. However, the novelty was the addition of adaptive terms in each one of the control loops. Although the closed-loop stability analysis was not included in the document, we were able to show convergence of the tracking error and boundedness of the adaptation parameters, which was done by using Lyapunov's theory and Barbalat's lemma. An original experimental platform was built and used to validate the proposed algorithm, which was compared experimentally with respect to a classical controller. Better results were obtained with the new controller, showing the advantages of using electrical and mechanical parameter adaptation.

Further research includes the consideration of other nonlinear effects in the system such as friction and saturation of the input voltage.

REFERENCES

[1] F. Khorrami, P. Krishnamurthy and H. Melkote, *Modeling and Adaptive Nonlinear Control of Electric Motors*, Sprinver–Verlag, Berlin, 2003.

[2] J. F. Gieras, *Permanent Magnet Motor Technology: Design and Applications*, CRC Press, Boca Raton, 2009.

[3] F. Blaschke, "The principle of field orientation as applied to the new transvector closed loop control for rotating field machines", Siemens Rev., vol. 39, pp. 217–220, May 1972.

[4] P. C. Krause, O. Wasynczuk, and S. D. Sudhoff, *Analysis of Electric Machinery and Drive Systems*, Piscataway, IEEE Press–Wiley, 3rd edition, 2013.

[5] W. Leonhard, *Control of Electrical Drives*, Springer–Verlag, Berlin, 2nd edition, 1996.

[6] Y. Sozer, Kaufman, H. and Torrey D., "Direct model reference adaptive control of permanent magnet brushless DC motor, In Proc. of the *1997 IEEE International Conference on Control Applications*, pp. 633-638, 1997.

[7] K. Kim, I. Baik, S. Chung and M. Youn, "Robust speed control of brushless DC motor using adaptive input–output linearisation technique", *IEE Proc.-Electr. Power Appl.*, Vol. 144, No. 6, pp. 469-475, 1997.

[8] H. Melkote and F. Khorrami, "Nonlinear adaptive control of direct-drive brushless DC motor and applications to robotic manipulators", *IEEE/ASME Transaction on Mechatronics*, Vol. 4, No. 1, pp. 71-81, 1999.

[9] Y. Lui and Zhu, Z., "Direct torque control of brushless DC drives with reduced torque ripple", *IEEE Transactions on Industry Applications* , Vol. 41 No.2, pp. 599-608, 2005.

[10] M. Ouassaid, M. Cherkaoui, A. Nejmi, and M. Maarou, "Nonlinear torque control for PMSM: A Lyapunov technique", in Proc. World Acad. Sci., Eng. Technol., vol. 6. Jun. 2005, pp. 118–121

[11] V. M. Hernández-Guzmán, V. Santibáñez and R. Campa, "PID control of robot manipulators equipped with brushless DC motors", *Robotica*, Vol. 27, No. 02, pp. 225–233, 2009.

[12] F. Mendoza-Mondragón, V. M. Hernández-Guzmán and R. V. Carrillo-Serrano, "Velocity Regulation in PMSMs Using Standard Field Oriented Control Plus Adaptation", *Asian Journal of Control*, Vol. 17, No. 6, pp. 2382–-2389, 2015.

[13] N. Hemati, J. S. Thorp and M. C. Leu, "Robust nonlinear control of brushless DC motors for direct-drive robotic applications", *IEEE Transactions on Industrial Electronics*, Vol. 37, No. 6, pp. 460-468, 1990.

[14] D. M. Dawson, J. Hu and T. Burg, *Nonlinear Control of Electronic Machinery*, Marcel Dekker, New York, 1998.

[15] R. V. Carrillo-Serrano, V. M. Hernández-Guzmán and V. Santibáñez, "PD control with feedforward compensation for rigid robots actuated by brushless DC motors", *Robotica*, Vol. 29, No. 4, pp. 507–514, 2011.

[16] S. Sastry and M. Bodson, *Adaptive control: Stability, Convergence and Robustness*, Prentice Hall, Englewood Cliffs, 1989.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 8, páginas 70 - 88.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

A master-slave synchronization of 3 dof robot manipulators based on super-twisting algorithm*

Jesús Martínez*, Omar Amaya, Oscar Salas**, Susana Gutiérrez**, Salvador A. Rodríguez*, Jesús de León****

* SEPI, ESIME Azcapotzalco, Instituto Politécnico Nacional, México City, México.

** Electrical Engineering Department, Autonomous University of Nuevo León, San Nicolás de los Garza, Nuevo León, México.

ing-ri-jmm@live.com.mx, xhomay@hotmail.com, salvador.sp@gmail.com, svgutierrez85@gmail.com, sarodriguezp@ipn.mx, drjleon@gmail.com

Abstract— **In this paper, a Super Twisting Algorithm control (STA) implemented for a Master-Slave Synchronization scheme on multi-robot system is presented. To address the problem of not having full access to the state vector one proposes to use a High-order Sliding Mode Differentiator, a virtual computational tool that is used for estimating the value of the speed from the position measurement which normally is available. Furthermore, the proposed control scheme increase robustness under non modeled dynamics, avoiding overestimate the gain. Experimental results are given to illustrate the performance of the proposed control scheme.**

I. INTRODUCTION

Controlled synchronization has become an important topic in recent years due to its application in teleoperation, manufacturing processes, systems working in coordinated schemes, assembly tasks, multi-robot systems, etc. The collaborative behavior provides flexibility and maneuverability that can hardly be obtained with a single system. An important field is the synchronization of robotic systems, [1-3].

In the formation of cooperative systems of robot manipulators, research has led to results in synthesis of multiple control laws. However, speaking of real applications, different problems related to physical constraints or cost arise. Therefore, it is important to analyze synchronization techniques that provide of maximum benefits and drawbacks experienced solve other control schemes.

Among the main limitations that are not fully resolved by some algorithms tracking control and master-slave synchronization, is that of having, in practical applications, only measurement vector of generalized positions (linear or angular displacement) and not speed, frequently forcing estimate this variable indirectly. It is resulting in a further complication of the control system. Another problem is having the ability to reject external disturbances while it has robustness against parametric uncertainties.

In the work discussed in [4], a controller that solves the problem of coordinating two (or more) robots, under a master-slave scheme in the case where only the position measurements are available is proposed. The controller consists of a feedback control law, and two observers nonlinear.

On the other hand, a simple decentralized continuous sliding PID controller for tracking tasks that yields semiglobal stability of all closed-loop signals with exponential convergence of tracking errors is proposed for robot manipulators in [5]. A dynamic sliding mode without reaching phase is

enforced, and terminal attractors, as well as saturated ones, are considered. Also, some authors have presented an original motion control strategy for robot manipulators based on the coupling of the inverse dynamics method with the so-called second-order sliding mode control approach, tested on an industrial manipulator, [6]. Using this method, in principle, all the coupling non-linearities in the dynamical model of the manipulator are compensated, transforming the multi-input non-linear system into a linear and decoupled one. Moreover, in [7], the super-twisting second-order sliding-mode algorithm is modified in order to design a velocity observer for uncertain mechanical systems. The finite time convergence of the observer is proved. Thus, the observer can be designed independently of the controller. A discrete version of the observer is considered and the corresponding accuracy is estimated.

This paper presents an approach of master-slave synchronization for a system of 4 DOF Robot Manipulators with homogeneous characteristics. Here, a variant of computed torque control is proposed. The main objective of the proposed controller is to overcome the parametric uncertainties by using Super-Twisting Algorithm (STA) [8, 9]. In fact, it has been shown in [10, 11] that this algorithm, which is based on the second order sliding mode technique, ensures robustness with respect to modeling errors and external disturbances while reducing the chattering phenomenon, caused by all first order sliding mode based controllers [12], (for the general problem of chattering see [13]). The stability and finite time convergence characteristics of the used algorithm have been recently proved by means of Lyapunov functions [14], so the stability analysis of the proposed controller has been conducted in the same way.

In order to show the effectiveness of the controller, experimental tests were carried out on experimental platform with two Robot Manipulators building in the Autonomous University of Nuevo León (UANL). The obtained results show the performance of the proposed controller in terms of stabilization, tracking and robustness with respect to parametric uncertainties.

The paper is organized as follows: In Section 2, the system description containing a mathematical model of Lagrangian multi-robot to define a master and slave system is presented. In Section 3 a Super-Twisting Control for tracking and synchronization master-salve control is derived. Moreover, in order to implement the controller, angular speed for the master and slave robot are estimated by High-order sliding mode Differentiator designed in Section 4. Experimental results given in Section 5, illustrate the effectiveness of the proposed scheme. Finally, Conclusions of this work are drawn.

II. DEFINITION OF LAGRANGIAN MULTI-ROBOT SYSTEM

In [1], [11], [12], using the Lagrangian formulation for its simplicity in dealing with complex systems involving multiple dynamics. The equations of motion for a robot with multiple joints ($\mathbf{q}_i \in \mathbb{R}^n$) can be derived by exploiting the Euler-Lagrange equations:

$$L_i = \frac{1}{2}\dot{\mathbf{q}}_i{}^T \mathbf{M}_i(\mathbf{q}_i)\dot{\mathbf{q}}_i - V_i; \quad \frac{d}{dt}\left[\frac{\partial L_i(\mathbf{q}_i,\dot{\mathbf{q}}_i)}{\partial \dot{\mathbf{q}}_i}\right] - \frac{\partial L_i(\mathbf{q}_i,\dot{\mathbf{q}}_i)}{\partial \mathbf{q}_i} = \boldsymbol{\tau}_i, \tag{1}$$

where ($1 \le i \le p$) denotes the index of robots or dynamic systems comprising a network, and p is the total number of the individual elements. Equation (1) can be represented as

$$\mathbf{M}_i(\mathbf{q}_i)\ddot{\mathbf{q}}_i + \mathbf{C}_i(\mathbf{q}_i,\dot{\mathbf{q}}_i)\dot{\mathbf{q}}_i + \mathbf{g}_i(\mathbf{q}_i) = \boldsymbol{\tau}_i, \tag{2}$$

where $\mathbf{M}_i(\mathbf{q}_i)$ is symmetric, positive definite and differentiable in $\mathbf{q}_i$, $\mathbf{g}_i(\mathbf{q}_i) = \frac{\partial V_i}{\partial \dot{\mathbf{q}}_i} \in \mathbb{R}^n$ and $\boldsymbol{\tau}_i$ is a generalized force or torque acting on the *i-th* robot (for this particular case, *i = m, s*). $\mathbf{C}_i(\mathbf{q}_i,\dot{\mathbf{q}}_i)\dot{\mathbf{q}}_i \in \mathbb{R}^n$ represents the Coriolis and centrifugal forces.

If the matrix $\mathbf{C}_i(\mathbf{q}_i,\dot{\mathbf{q}}_i)\dot{\mathbf{q}}_i \in \mathbb{R}^n$ is defined using the Christoffel symbols, then the matrix $\dot{\mathbf{M}}_i(\mathbf{q}_i) - \mathbf{2C}_i(\mathbf{q}_i,\dot{\mathbf{q}}_i)$ is skew-symmetric *i.e.*

$$\boldsymbol{x}^T\left(\dot{\mathbf{M}}_i(\mathbf{q}_i) - \mathbf{2C}_i(\mathbf{q}_i,\dot{\mathbf{q}}_i)\right)\boldsymbol{x} = 0; \quad \forall \boldsymbol{x} \epsilon \mathbb{R}^n, \tag{3}$$

The matrices $\mathbf{M}_i(\mathbf{q}_i)$ and $\mathbf{C}_i(\mathbf{q}_i,\dot{\mathbf{q}}_i)$ are bounded with respect to $\mathbf{q}_i$, as

$$0 < \mathbf{M}_{i,m} \le \|\mathbf{M}_i(\mathbf{q}_i)\| \le \mathbf{M}_{i,M}; \quad \forall \boldsymbol{x} \epsilon \mathbb{R}^n, \tag{4}$$

$$\|\mathbf{C}_i(\mathbf{q}_i, x)\| \le \mathbf{C}_{i,M}\|\boldsymbol{x}\|; \qquad \forall \boldsymbol{x} \epsilon \mathbb{R}^n, \tag{5}$$

For this case, we define the master system as

$$\mathbf{M}_{\boldsymbol{m}}(\mathbf{q}_m)\ddot{\mathbf{q}}_m + \mathbf{C}_{\boldsymbol{m}}(\mathbf{q}_m, \dot{\mathbf{q}}_m)\dot{\mathbf{q}}_m + \mathbf{g}_m(\mathbf{q}_m) = \boldsymbol{\tau}_m, \tag{6}$$

and the slave system is

$$\mathbf{M}_{\boldsymbol{s}}(\mathbf{q}_s)\ddot{\mathbf{q}}_s + \mathbf{C}_{\boldsymbol{s}}(\mathbf{q}_s, \dot{\mathbf{q}}_s)\dot{\mathbf{q}}_s + \mathbf{g}_s(\mathbf{q}_s) = \boldsymbol{\tau}_s. \tag{7}$$

The physical parameters of the links in the robotic manipulators are described in Table I, these were estimated using the software Solidworks®. While the inertia matrices of inertia are known, it is important to mention that we do not know the coriolis matrices exactly, therefore we have parametric uncertainty.

III. SYNCHRONIZATION BASED ON STA

In master-slave synchronization, there are two important control tasks. The first is to propose a tracking control strategy that allows the master system to follow a desired trajectory. In addition, the second is to propose synchronization control strategy that allows the slave system to achieve a master trajectory.

Therefore, for unidirectional schemes as in this case, it made the following assumptions:

Assumption 1. There is not any feedback from the slave to the master. Control can be performed only in one direction.

Assumption 2. The master-slave synchronization occurs no matter how initialized them. It must be that their states asymptotically coincide, which call it synchronization criterion, *i.e.*

$$\lim_{t\to\infty} \left\| \begin{pmatrix} \hat{\mathbf{q}}_m \\ \dot{\hat{\mathbf{q}}}_m \end{pmatrix} - \begin{pmatrix} \mathbf{q}_s \\ \dot{\hat{\mathbf{q}}}_s \end{pmatrix} \right\| = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{8}$$

After considerate the above assumptions, it proceeds to write the equation (6) as follows

$$\ddot{\mathbf{q}}_m = \mathbf{M}_{\boldsymbol{m}}(\mathbf{q}_m)^{-\mathbf{1}}[\boldsymbol{\tau}_m - \mathbf{C}_m(\mathbf{q}_m, \dot{\mathbf{q}}_m)\dot{\mathbf{q}}_m - \mathbf{g}_m(\mathbf{q}_m)], \tag{9}$$

where $\mathbf{q}_m \in \mathbb{R}^n$ is a vector representing the positions or generalized coordinates and $\dot{\mathbf{q}}_m$ represents the velocity vector. It is important to note that vector $(\mathbf{q}_m \quad \dot{\mathbf{q}}_m)^T$ represents the states of the master system. A tracking control law that eliminates nonlinearities of the equation (9), using state feedback, is given by

$$\boldsymbol{\tau}_m = \mathbf{M}_{\boldsymbol{m}}(\mathbf{q}_m)\boldsymbol{\nu}_m + \mathbf{C}_m(\mathbf{q}_m, \dot{\mathbf{q}}_m)\dot{\mathbf{q}}_m + \mathbf{g}_m(\mathbf{q}_m), \tag{10}$$

where $\boldsymbol{\nu}_m$ is a new "auxiliary input". In the equation (10) the terms $\mathbf{C}_m(\mathbf{q}_m, \dot{\mathbf{q}}_m)\dot{\mathbf{q}}_m + \mathbf{g}_m(\mathbf{q}_m)$, are added in order to cancel from equation (9). Therefore, the system obtained is

$$\mathbf{M}_{\boldsymbol{m}}(\mathbf{q}_m)\ddot{\mathbf{q}}_m = \mathbf{M}_{\boldsymbol{m}}(\mathbf{q}_m)\boldsymbol{\nu}_m. \tag{11}$$

Because $\mathbf{M}_{\boldsymbol{m}}(\mathbf{q}_m)$ is invertible, from (11) a linear closed loop system is defined as

$$\ddot{\mathbf{q}}_m = \boldsymbol{\nu}_m, \tag{12}$$

which is in the Brunovsky canonical form [13].

The auxiliary input $\boldsymbol{\nu}_m$ is designed by a robust control strategy based on second order sliding modes known as Super Twisting Algorithm (STA) [9] and [11]. As mentioned in [18] and [19], its goal is to enforce the sliding mode on the manifold

$$\boldsymbol{\sigma}_m = \boldsymbol{\lambda}_{m1}\tilde{\mathbf{q}}_m + \boldsymbol{\lambda}_{m2}\dot{\tilde{\mathbf{q}}}_m, \tag{13}$$

where $\tilde{\mathbf{q}}_m = \mathbf{q}_m - \mathbf{q}_d$ and $\dot{\tilde{\mathbf{q}}}_m = \dot{\hat{\mathbf{q}}}_m - \dot{\mathbf{q}}_d$, $\mathbf{q}_d$ and $\dot{\mathbf{q}}_d$ are the desired position and desired velocity respectively, $\boldsymbol{\lambda}_{m1}$ and $\boldsymbol{\lambda}_{m2}$ are positive diagonal matrices. The implementation of the control scheme (10) requires knowledge of the master robot dynamic model, which is important to mention that one of the benefits of STA is its tolerance for parametric uncertainties. $\dot{\mathbf{q}}_m$ is other necessary element, which will be estimated by a robust differentiator (see Section 4) that gotten $\dot{\hat{\mathbf{q}}}_m$, because it doesn't have the complete information of the state vector. Therefore, equation (10) is written as

$$\boldsymbol{\tau}_m = \mathbf{M}_{\boldsymbol{m}}(\mathbf{q}_m)\boldsymbol{\nu}_m + \mathbf{C}_m\left(\mathbf{q}_m, \dot{\hat{\mathbf{q}}}_m\right)\dot{\hat{\mathbf{q}}}_m + \mathbf{g}_m(\mathbf{q}_m), \tag{14}$$

Considering the previous, $\boldsymbol{\nu}_m$ is defined as

$$\begin{aligned} \boldsymbol{\nu}_m &= -\boldsymbol{K}_{m1}|\boldsymbol{\sigma}_m|^{1/2} sign(\boldsymbol{\sigma}_m) + \boldsymbol{\mu}_m, \\ \dot{\boldsymbol{\mu}}_m &= -\boldsymbol{K}_{m2} sign(\boldsymbol{\sigma}_m), \end{aligned} \tag{15}$$

where $\boldsymbol{K}_{m1}$ and $\boldsymbol{K}_{m2}$ are positive diagonal matrices and $\boldsymbol{\sigma}_m$ is the sliding surface for tracking control which it is defined in (13). The STA control (15) is continuous since both $-\boldsymbol{K}_{m1}|\boldsymbol{\sigma}_m|^{1/2}$ and the term $\boldsymbol{\mu}_m = -\boldsymbol{K}_{m2}\int sign(\boldsymbol{\sigma}_m)$ are continuous. Now, the high frequency switching term $sign(\boldsymbol{\sigma}_m)$ is "hidden" under the integral [9].
In addition to equation (8) is met, a strategy for master-slave synchronization is designed below.

Now equation (6) is written as follow

$$\ddot{\mathbf{q}}_s = \mathbf{M}_s(\mathbf{q}_s)^{-1}[\boldsymbol{\tau}_s - \mathbf{C}_s(\mathbf{q}_s, \dot{\mathbf{q}}_s)\dot{\mathbf{q}}_s - \mathbf{g}_s(\mathbf{q}_s)], \tag{16}$$

where $\mathbf{q}_s \in \mathbb{R}^n$ is a vector representing the positions or generalized coordinates from slave robot and $\dot{\mathbf{q}}_s$ represents the velocity vector. It is important to note that vector $(\mathbf{q}_s \quad \dot{\mathbf{q}}_s)^T$ represents the states of the slave system. Accordingly a control law that synchronizes the slave system to the master, *i.e.* the slave can achieve the master trajectories $\mathbf{q}_m$ and $\dot{\mathbf{q}}_m$, is given by

$$\boldsymbol{\tau}_s = \mathbf{M}_s(\mathbf{q}_s)\boldsymbol{\nu}_s + \mathbf{C}_s(\mathbf{q}_s, \dot{\mathbf{q}}_s)\dot{\mathbf{q}}_s + \mathbf{g}_s(\mathbf{q}_s), \tag{17}$$

where $\boldsymbol{\nu}_s$ is a new "auxiliary input". In the equation (17) the terms $\mathbf{C}_s(\mathbf{q}_s, \dot{\mathbf{q}}_s)\dot{\mathbf{q}}_s + \mathbf{g}_s(\mathbf{q}_s)$, are added in order to cancel from equation (16). Therefore, the system obtained is

$$\mathbf{M}_s(\mathbf{q}_s)\ddot{\mathbf{q}}_s = \mathbf{M}_s(\mathbf{q}_s)\boldsymbol{\nu}_s. \tag{18}$$

Because $\mathbf{M}_s(\mathbf{q}_s)$ is invertible, from (18) a linear closed loop system is defined as

$$\ddot{\mathbf{q}}_s = \boldsymbol{\nu}_s. \tag{19}$$

It is not easy to see that the master-slave synchronization is given, however this is achieved by means of the following *synchronization sliding surface*

$$\boldsymbol{\sigma}_s = \boldsymbol{\lambda}_{s1}\tilde{\mathbf{q}}_s + \boldsymbol{\lambda}_{s2}\dot{\tilde{\mathbf{q}}}_s, \tag{20}$$

where $\tilde{\mathbf{q}}_s = \mathbf{q}_s - \hat{\mathbf{q}}_m$ and $\dot{\tilde{\mathbf{q}}}_s = \dot{\hat{\mathbf{q}}}_s - \dot{\hat{\mathbf{q}}}_m$, $\hat{\mathbf{q}}_m$ and $\dot{\hat{\mathbf{q}}}_m$ are the master estimated position and master estimated velocity respectively, $\boldsymbol{\lambda}_{s1}$ and $\boldsymbol{\lambda}_{s2}$ are positive diagonal matrices defined. The implementation of the control scheme (17) with the synchronization surface (20), requires knowledge of the slave robot dynamic model, $\hat{\mathbf{q}}_m$, $\dot{\hat{\mathbf{q}}}_m$ and $\dot{\hat{\mathbf{q}}}_s$ are other necessary elements, which will be estimated by a robust differentiator (see Section 4). Then, equation (17) is written as

$$\boldsymbol{\tau}_s = \mathbf{M}_s(\mathbf{q}_s)\boldsymbol{\nu}_s + \mathbf{C}_s\left(\mathbf{q}_s, \dot{\hat{\mathbf{q}}}_s\right)\dot{\hat{\mathbf{q}}}_s + \mathbf{g}_s(\mathbf{q}_s), \tag{21}$$

Considering the previous, $\boldsymbol{\nu}_s$ is defined as

$$\begin{aligned}\boldsymbol{\nu}_s &= -\boldsymbol{K}_{s1}|\boldsymbol{\sigma}_s|^{1/2} sign(\boldsymbol{\sigma}_s) + \boldsymbol{\mu}_s,\\ \dot{\boldsymbol{\mu}}_s &= -\boldsymbol{K}_{s2} sign(\boldsymbol{\sigma}_s),\end{aligned} \tag{22}$$

where $\boldsymbol{K}_{s1}$ and $\boldsymbol{K}_{s2}$ are positive diagonal matrices and $\boldsymbol{\sigma}_s$ is the sliding surface for synchronization which it is defined in (20). The proof of finite time convergence to zero of the variables $\boldsymbol{\nu}_m$ and $\boldsymbol{\nu}_s$ is explicitly given in an Appendix. It is conducted in the same way that of [14]. Therefore, it can be concluded that the synchronization criterion (8) is fulfilled.

IV. HIGH-ORDER SLIDING MODE DIFFERENTIATOR

A. Levant proposed in [20] and [21], a robust high-order differentiator based on sliding mode as follows.
Suppose that it is known that the input signal is compounded of a smooth signal $f_0(t)$ to be differentiated and a noise being a bounded Lebesgue measurable function time. Both signals are unknown and only their sum is available. It is proved that if the base signal $f_0(t)$ has *r-th* derivate with Lipchitz's constant *L>0*, the best possible *k-th* order differentiation accuracy is $d_k L^{k/(r+1)} \varepsilon^{(r-k+1)/(r+1)}$, where $d_k > 1$ may be estimated (this asymptotic representation may be improved with additional restrictions on $f_0(t)$. Moreover, it is proved that such a robust exact differentiator really exists.

Let the aim be to find real-time robust estimations of $f_0(t), \dot{f}_0(t), \dots, f_0^{(r)}(t)$, being exact in the absence of measurement noise and continuously depending on the noise magnitude. The

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

differentiator is recursively constructed. Let a $(r-1)th-order$ differentiator $D_{r-1}(f(t),L)$, produce outputs D_{r-1}^{i} $(i=0,1,\dots,r-1)$, which are estimates of $f_0(t),\dot{f}_0(t),\dots,f_0^{(r)}(t)$ for any input signal f with $f_0^{(r-1)}$ having Lipschitz constant $L>0$. Then, the *r-th* order differentiator has the outputs $z_i=D_r^i,\ \ i=0.1,\dots r$, defined as follows

$$\begin{aligned}&\dot{z}_0=\eta,\ \ \eta=-\varrho L^{1/r}|z_0-f(t)|^{\frac{r-1}{r}}sign\big(z_0-f(t)\big)+z_1,\\&z_1=D_{r-1}^{0}(\eta,L),\ \ \dots,z_1=D_{r-1}^{r-1}(\eta,L).\end{aligned}\tag{23}$$

Here $D_{r-1}(f(t),L)$ is a simple nonlinear filter

$$D_0:\dot{z}=-\varrho Lsign\big(z_0-f(t)\big);\quad \varrho<L.\tag{24}$$

In other words the *r-th* order differentiator has the form

$$\begin{aligned}&\dot{z}_0=\eta_0,\ \ \eta_0=-\varrho_0L^{1/r}|z_0-f(t)|^{\frac{r-1}{r}}sign\big(z_0-f(t)\big)+z_1,\\&\dot{z}_1=\eta_1,\ \ \eta_1=-\varrho_1L^{1/r-1}|z_1-\eta_0|^{\frac{r-2}{r-1}}sign(z_1-\eta_0)+z_2,\\&\dots\\&\dot{z}_{r-2}=\eta_{r-2},\ \ \eta_{r-2}=-\varrho_{r-2}L^{1/2}|z_{r-2}-\eta_{r-3}|^{\frac{1}{2}}sign(z_{r-2}-\eta_{r-3})+z_{r-1},\\&\dot{z}_0=-\varrho_{r-1}Lsign\big(z_{r-1}-\eta_{r-2}\big).\end{aligned}\tag{25}$$

Taking into account the above, we can define the structure of the robust differentiator to estimate $\hat{\mathbf{q}}_m$, $\dot{\hat{\mathbf{q}}}_m$ and $\dot{\hat{\mathbf{q}}}_s$

$$\begin{aligned}&\dot{\hat{q}}_{ji,0}=\eta_{ji,0},\\&\eta_{ji,0}=-\varrho_0L^{1/3}\big|\hat{q}_{ji,0}-q_{ji}\big|^{\frac{2}{3}}sign\big(\hat{q}_{ji,0}-q_{ji}\big)+\hat{q}_{ji,1},\\&\dot{\hat{q}}_{ji,1}=\eta_{ji,1},\\&\eta_{ji,1}=-\varrho_1L^{1/2}\big|\hat{q}_{ji,1}-\eta_{ji,0}\big|^{\frac{1}{2}}sign\big(\hat{q}_{ji,1}-\eta_{ji,0}\big)+\hat{q}_{ji,2},\\&\dot{\hat{q}}_{ji,2}=-\varrho_2Lsign\big(\hat{q}_{ji,2}-\eta_{ji,1}\big),\quad j=\,m,s,\quad and\ \ i=1,2,3,\end{aligned}\tag{26}$$

where q_{ji} for $j=m,s$ and $i=1,2,3$; is the output measurable, The position estimation error is $\bar{q}_{ji,0}=\hat{q}_{ji,0}-q_{ji}$ and $\eta_{ji,0}$ is the estimation speed for each joint.

Proposition 1. Consider the system (6) and (7) in closed-loop with the controller (14) -(21), using the estimates obtained by the differentiator (26). Then, the trajectories of the slave system (7) converge in finite-time to the trajectories of the master system (6).

V. EXPERIMENT

A. HARDWARE AND SOFTWARE

In order to validate the proposed controller, experimental tests were carried out in an experimental platform which consist in two homogeneous Robot Manipulators building in the UANL, each robot has four degree of freedom of rotational configuration, three of them describe the principal movements of the robot and define his workspace, the last one DOF describe the orientation of the gripper, the end effector always stay parallel in the XY plane. The actuators are servomotors modified which only consist in a motor of DC coupled to a gear train, and a potentiometer to measure the articular position of the links of the robot (see “Figure 1”).

For the acquisition and transmission of information an Arduino® Due was used, that consist in a microcontroller board based on the Atmel® SAM3X8E ARM Cortex-M3 CPU. It has 54 digital input/output pins (of which 12 can be used as PWM outputs), 12 analog inputs, 4 UARTs (hardware serial ports), an 84 MHz clock, an USB OTG capable connection, 2 DAC (digital to analog), 2 TWI, a power jack, an SPI header, a JTAG header, a reset button and an erase button.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

The control law given by Eq. (14) and synchronization law given by Eq. (21) were developed on a PC Intel Core i3. Control scheme were developed in MATLAB/Simulink® environment, because there are special blocks for transmission and acquisition signal for Arduino® board. Using Runge-Kutta solver with an integration step of 0.005 s., was implemented this control. In Figure 2 it can observed as communicating with sensors and actuators is achieved through card Arduino® Due and MATLAB/Simulink®. The physical parameters of the Robots are given in Table I with reference to Figure 3.

It is important to mention that the algorithm and the plant were programed in continuous time, only the acquisition and sending of the information was made in a discrete way, so that an analysis of the discretization not was considered, but nevertheless, would be an important point for a future work.

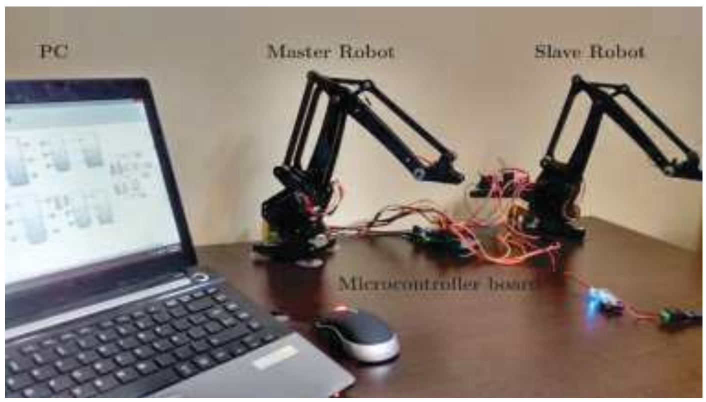


Figure 1. Experimental Platform. Consists of two Robot Manipulators, an Arduino Due board and a PC with MATLAB/Simulink®.

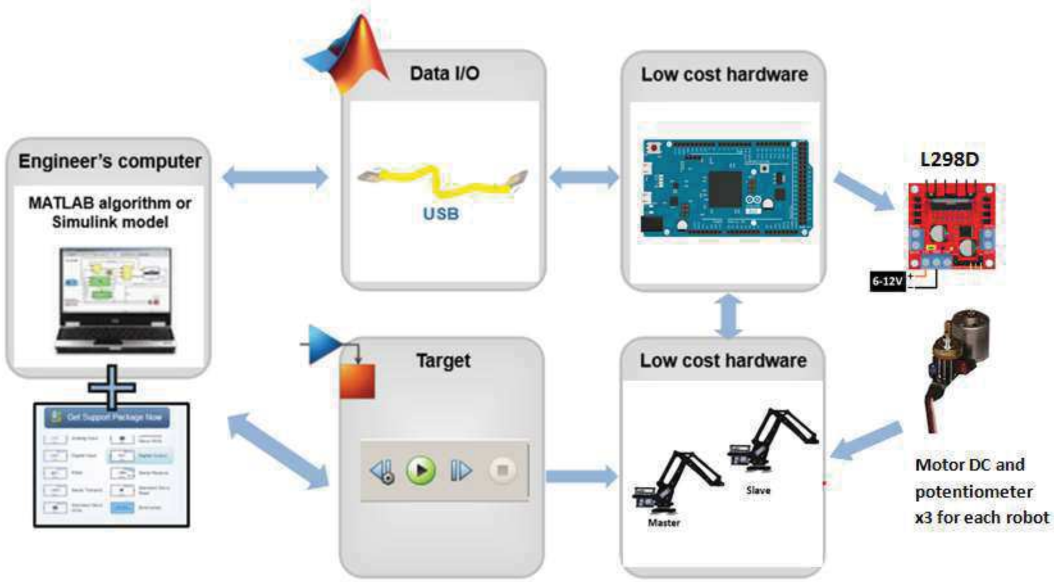


Figure 2. Diagram, which showing, how the transmission and acquisition data system between hardware and software is done.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

TABLE I. Parameters of the Robots

Parameter	Notation	Value
Length of the base	l_b	0.09277 m
Length 1=3	$l_1= l_3$	0.148 m
Length2=4	$l_2= l_4$	0.054 m
Length 5	l_5	0.16 m
Length 6	l_6	0.0606 m
Mass base	m_b	0.33 kg
Mass 1	m_1	0.051 kg
Mass 2	m_2	0.02 kg
Mass 3	m_3	0.01 kg
Mass 4	m_4	0.05 kg
Mass 5	m_5	0.03 kg
Center of mass 1	l_{c1}	0.091 m
Center of mass 2	l_{c2}	0.048 m
Inertia base	I_b	0.01 kg.m^2
Inertia 1	I_1	2.35x10^{-3} kg.m^2
Inertia 2	I_2	2.30x10^{-6} kg.m^2
Inertia 3	I_3	0.71x10^{-3} kg.m^2
Inertia 4	I_4	1.05x10^{-3} kg.m^2

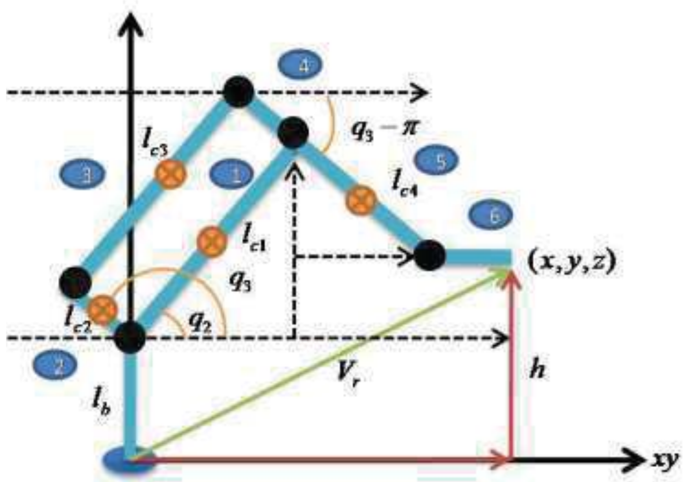


Figure 3. Profile view of the four DOF Robot Manipulator.

B. EXPERIMENTAL CONDITIONS

In order that the robot has soft movements according to the constraints of the actuators, a polynomial functions class C^n was proposed to generate the desired trajectories for each joint. The function is

$$q_{di} = a_3t^3 + a_2t^2 + a_1t + a_0; \quad i = 1,2,3, \tag{27}$$

where $a_0 = q_{0i}$, $a_1 = 0$, $a_2 = \left(\frac{3}{t_s^2}\right)(q_{fi} - q_{0i})$, $a_3 = -\left(\frac{2}{t_s^3}\right)(q_{fi} - q_{0i})$, q_{0i} is the initial position and q_{fi} is the final position.

For this application, the chosen gains for STA are given as

$$\boldsymbol{\lambda}_{m1} = \begin{pmatrix} 5 & 0 & 0 \\ 0 & 4.2 & 0 \\ 0 & 0 & 5 \end{pmatrix}, \quad \boldsymbol{\lambda}_{m2} = \begin{pmatrix} 0.02 & 0 & 0 \\ 0 & 0.0327 & 0 \\ 0 & 0 & 0.0045 \end{pmatrix}, \tag{28}$$

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

$$\boldsymbol{\lambda}_{s1} = \begin{pmatrix} 8 & 0 & 0 \\ 0 & 4.2 & 0 \\ 0 & 0 & 4.2 \end{pmatrix}, \quad \boldsymbol{\lambda}_{s2} = \begin{pmatrix} 0.02 & 0 & 0 \\ 0 & 0.003 & 0 \\ 0 & 0 & 0.001 \end{pmatrix}, \tag{29}$$

$$\boldsymbol{K}_{m1} = \begin{pmatrix} 4 & 0 & 0 \\ 0 & 4.5 & 0 \\ 0 & 0 & 3.5 \end{pmatrix}, \quad \boldsymbol{K}_{m2} = \begin{pmatrix} 0.35 & 0 & 0 \\ 0 & 0.30 & 0 \\ 0 & 0 & 0.15 \end{pmatrix}, \tag{30}$$

$$\boldsymbol{K}_{s1} = \begin{pmatrix} 3 & 0 & 0 \\ 0 & 3.4 & 0 \\ 0 & 0 & 4 \end{pmatrix}, \quad \boldsymbol{K}_{s2} = \begin{pmatrix} 0.12 & 0 & 0 \\ 0 & 0.30 & 0 \\ 0 & 0 & 0.027 \end{pmatrix}. \tag{31}$$

High-order sliding mode differentiator gains are displayed on Table II.

TABLE II. High-order sliding mode differentiator gains for Master and Slave Robots

Joint	ϱ_0	ϱ_1	ϱ_2	L
q_{ji}	127.5	105	7.5	1

The tracking control strategy (14) is proposed for the master robot in order that this reaches the desired trajectories and synchronization control strategy (21) is implemented for the slave robot that tries to follow master trajectories. The desired trajectories are three different continuous signals constructed from a third order polynomial designed considering the workspace of the robots.

C. Experimental results

For a better understanding of the plots, the Figures 4-11 are shown in three windows. Each window shows the performance of each of the joints in the time interval $[0,7)$ $sec.$, during the master-slave synchronization experiment.

From responses in Figure 4, it can be seen that master-slave synchronization scheme is fulfilled. In addition, here we can see that the estimated trajectories for the master robot not fully reach the desired trajectories so a tracking error of the joint 1 to 2 and the joint 2 to 3 is induced. This effect is also reflected in the slave robot as shown in Figure 5. It should be recalled that the main objective in synchronization systems is that slaves reach the trajectories of the master, but is not necessarily that the master reaches the desired trajectories. Notice that the goal is to follow the trajectories of the master robot $\hat{\boldsymbol{q}}_m$, and not the desired master robot trajectories $\boldsymbol{q}_d$ which may not be achieved by the master because of noise, parametric uncertainty or unmodeled dynamics of the master robot, like unmodeled friction phenomena, unknown loads, etc., [27].

In Figure 5, synchronization errors for each of the generalized coordinates are presented. The window 3 shows that the synchronization error is larger in the third joint of the slave, this is induced the link 2 to 3.

Figure 6 depicts the estimated angular positions of the master robot. This shows how the existing noise in the measured signals is reduced considerably with the estimated signals.

The estimation error for each master joint coordinates is in Figure 7. The window 3 shows that the estimation error is larger in the third joint of the slave, this is induced the link 2 to 3.

Estimated angular velocities of master and slave are shown in Figures 8 and 9, respectively. The experimental results present errors bound in the transitory state (see Figure 4), this effect is observed mainly in the estimated velocity (see Figure 8 and 9), possibly due to noise amplification and unmolded dynamics.

Finally, the control inputs for the master and slave are shown in Figures 10 and 11 respectively.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

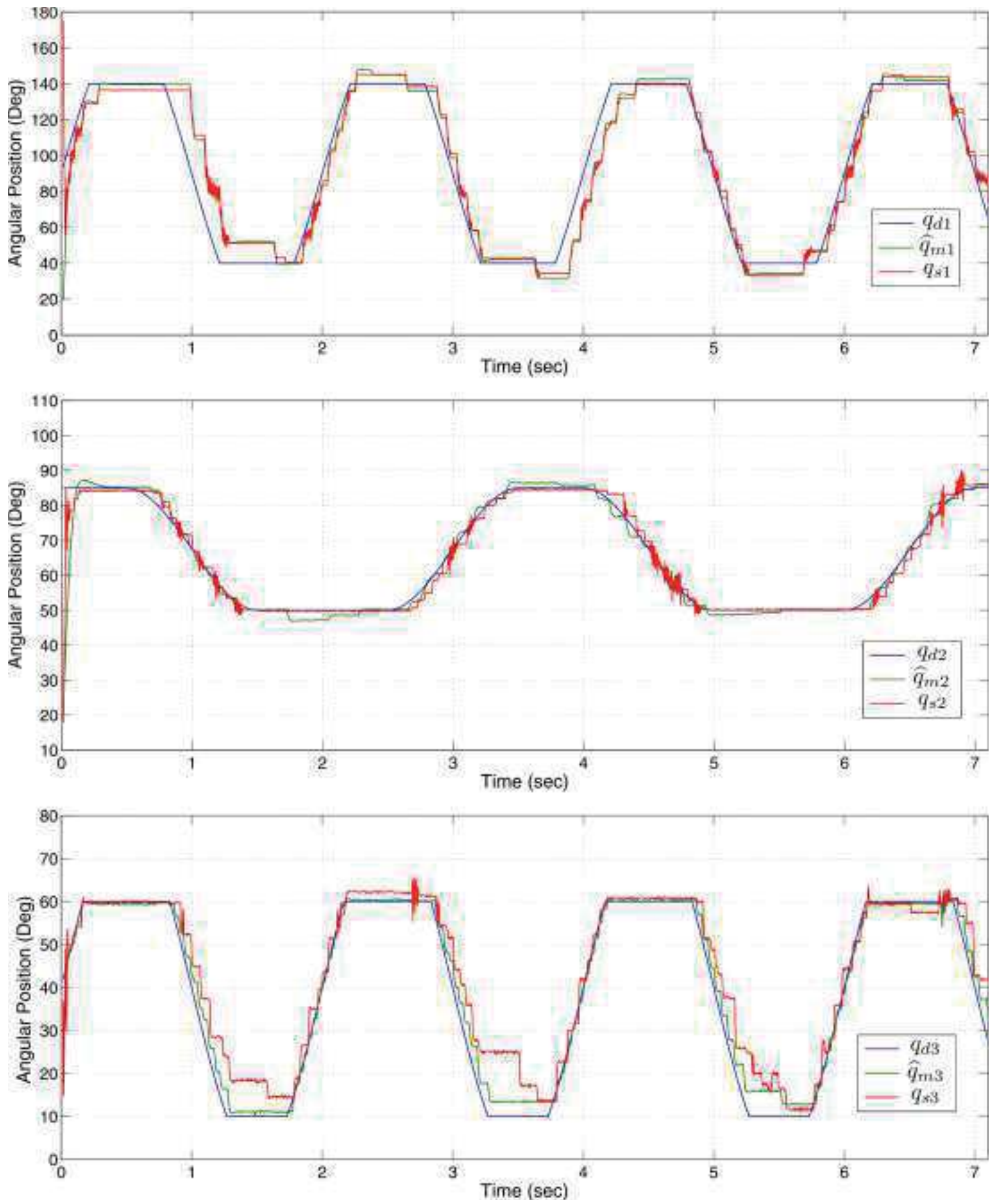


Figure 4. Master-slave Synchronization for each joint coordinates. Desired trajectories for each joint are bounded periodic signals these are in blue; the estimated trajectories for the master robot are in green and the measures trajectories of the slave robot are in red.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

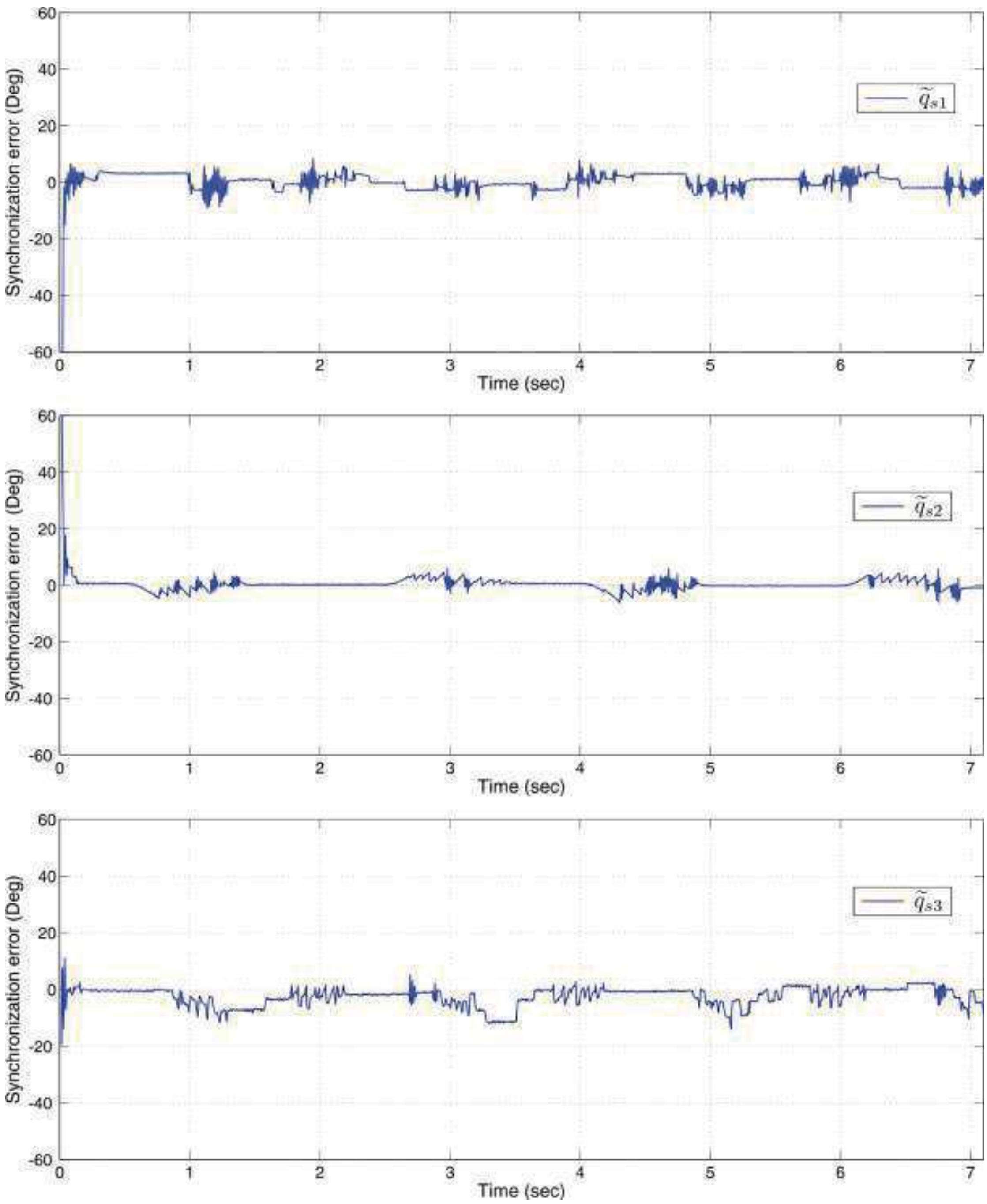


Figure 5. Synchronization errors. The average error in the time interval $[0,7)$ sec. for $\tilde{q}_{s1}$ is 0.0622, of $\tilde{q}_{s2}$ is 0.279 and to $\tilde{q}_{s3}$ is 2.352.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

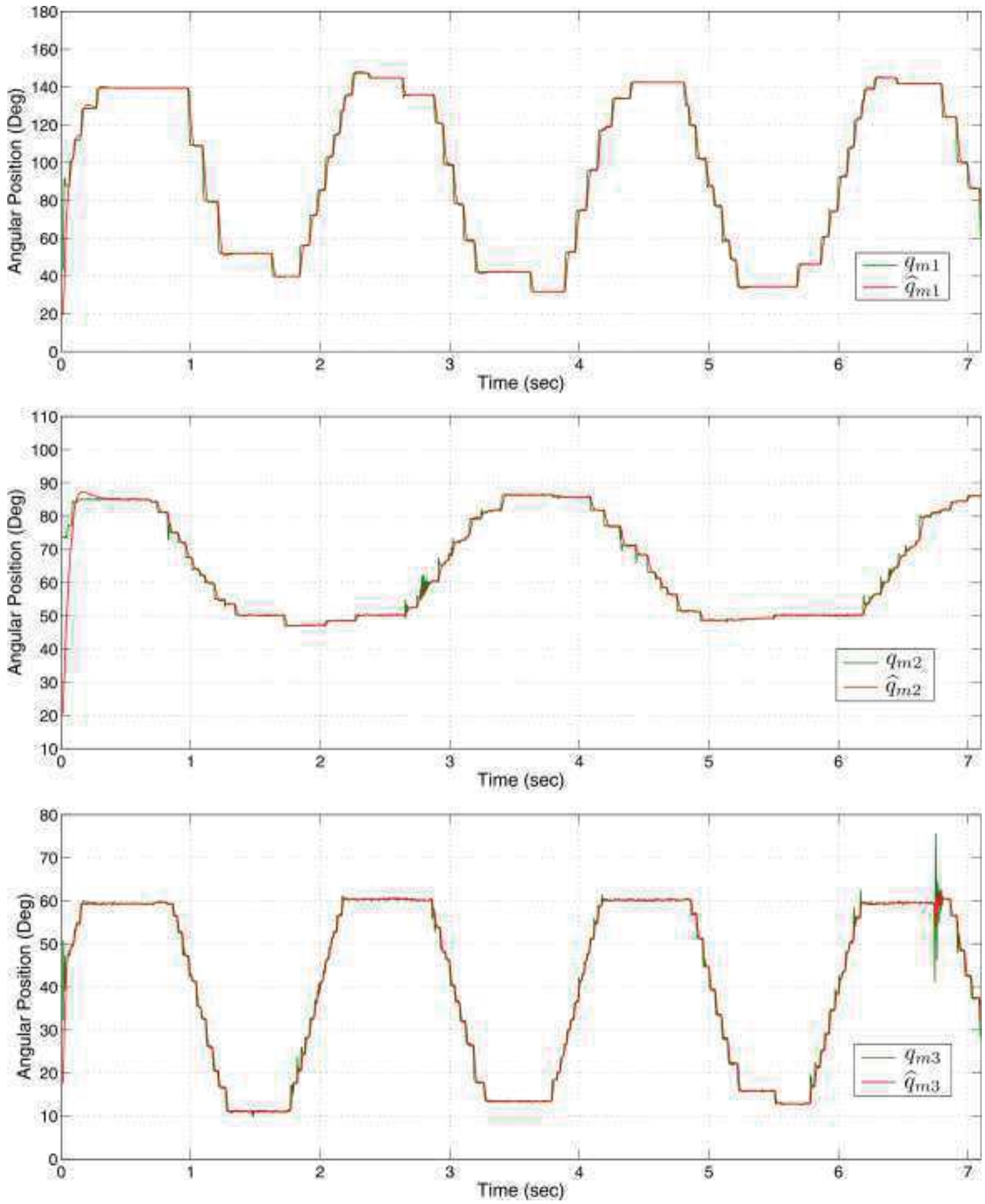


Figure 6. Angular positions $\mathbf{q}_m$ and estimated angular positions $\widehat{\mathbf{q}}_m$ of the Master Robot. Measures master robot positions are green and estimated positions are red.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 8, páginas 70 - 88.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

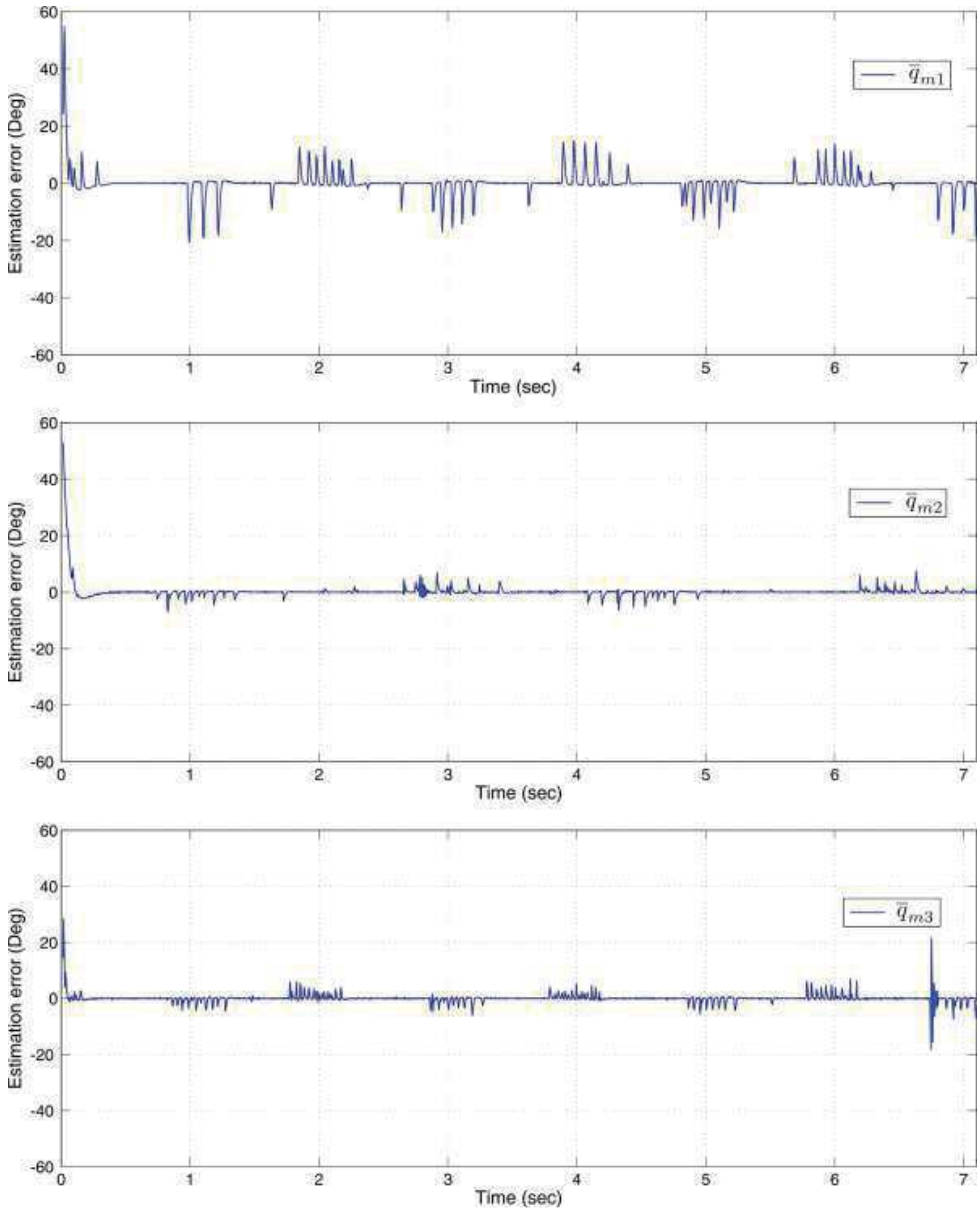


Figure 7. Estimation errors for each master joint coordinates. The average error in the time interval $[0,7)$ sec. for $\bar{q}_{m1}$ is 0.0054, of $\bar{q}_{m2}$ is 0.085 and to $\bar{q}_{m3}$ is 0.0059.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

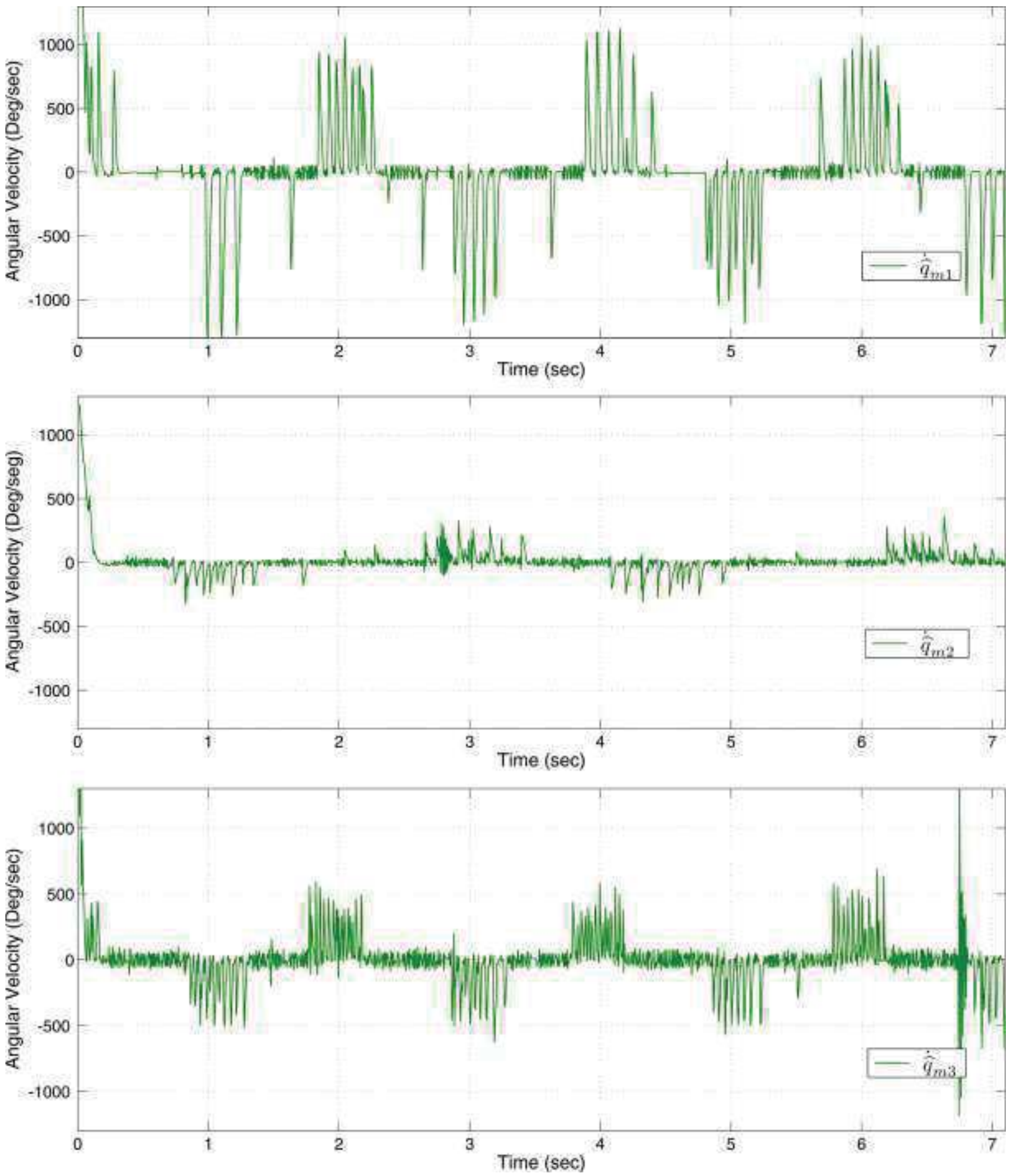


Figure 8. Estimated angular velocities of Master Robot.

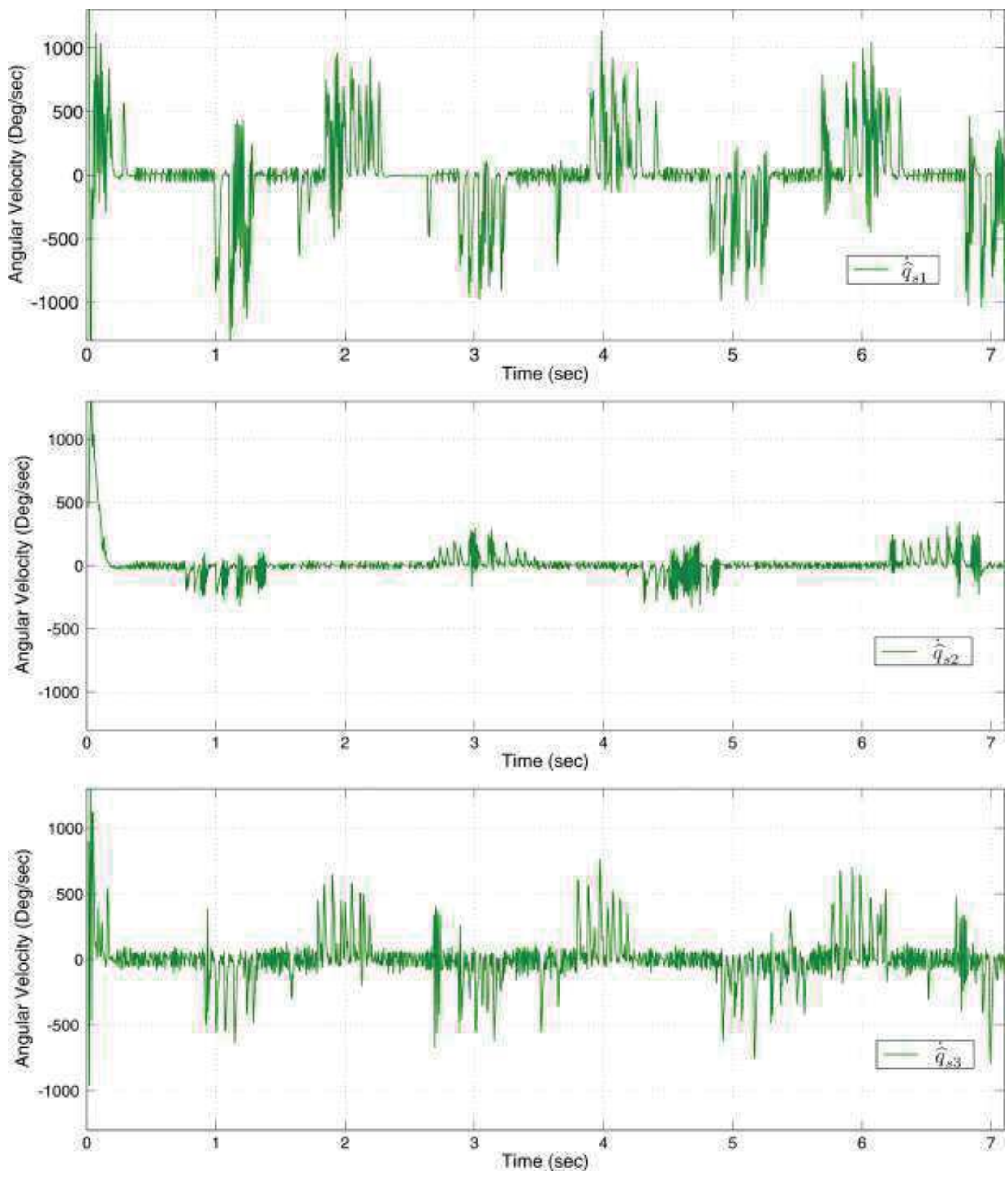


Figure 9. Estimated angular velocities of Slave Robot.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

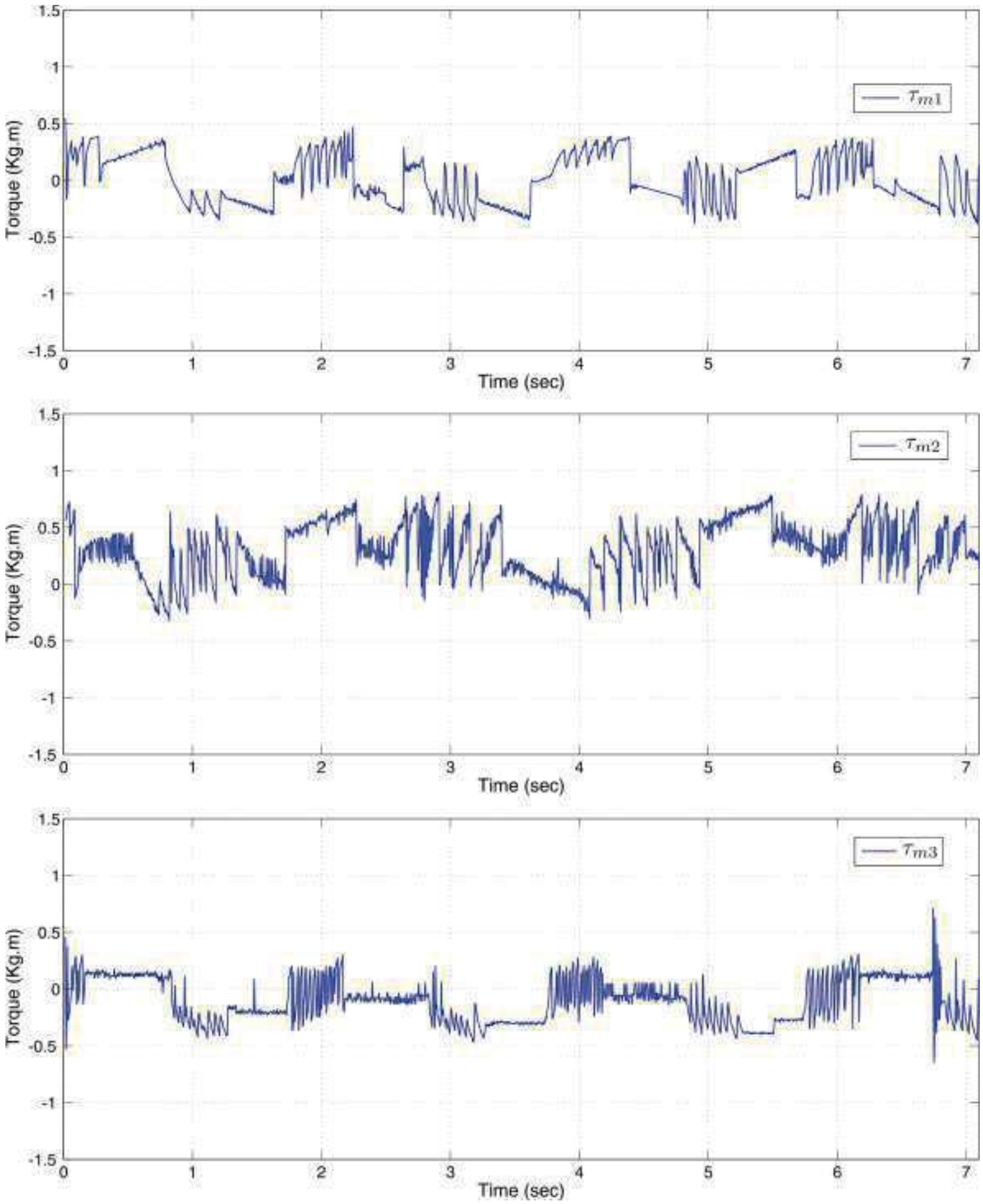


Figure 10. Control inputs for the Master Robot.

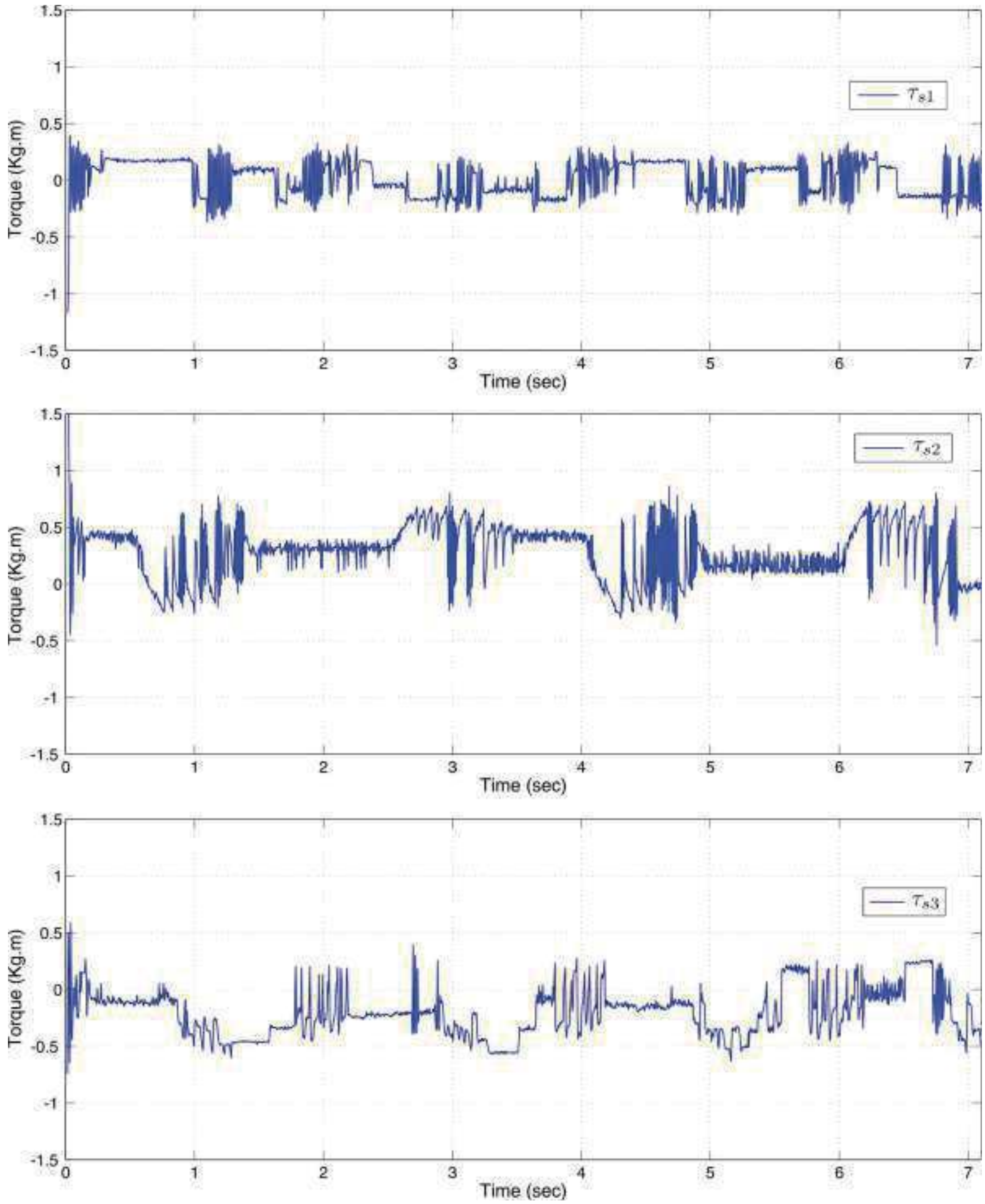


Figure 11. Control inputs for the Slave Robot.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

VI. CONCLUSION

In this paper, a super twisting controller algorithm has been proposed and successfully implemented on a Master-Slave Robot System. The controller has been designed using second order sliding mode approach in order to avoid chattering phenomenon and to ensure robustness with respect to model uncertainties. With the aim of implementing the proposed controller, a High-order sliding mode differentiator was designed for estimating non-measurable states. The experimental results obtained on a Master-Slave Robot System clearly show the effectiveness of the proposed controller in the Synchronization case.

Although, several authors have implemented in nonlinear dynamical systems and robotic manipulators, controllers based on second order sliding mode as the STA efficiently, the greatest contribution of this work is the modification of a classic control such as the computed-torque control which is generally designed with a PD control in closed loop. Its controller was replaced by the STA on a Master-Slave Robot scheme.

Finally, because the STA controller is implemented with the high order differentiator for estimating the angular velocities, tuning is facilitated. Contrary to the implementation of some classic controllers as presented in [26], since high order sliding mode differentiator (observer) converges in finite-time, control law and observer can be designed separately, *i.e.*, separation principle is satisfied.

APPENDIX

A. LYAPUNOV STABILITY ANALYSIS OF THE STA

Consider the standard Super-Twisting Algorithm (STA) without perturbation term

$$\begin{aligned} \dot{z}_1 &= -K_1|z_1|^{1/2} sign(z_1) + z_2, \\ \dot{z}_2 &= -K_2 sign(z_1), \end{aligned} \tag{32}$$

where z_i, $i = 1,2$, are the scalar state variables, $-K_i$ are positive gains to be designed. The solutions are all trajectories in the sense of Filippov [21]. The proof is stated as in [23] and [24]. Consider the Lyapunov function presented in [24]

$$V(z) = \varsigma^T P \varsigma, \tag{33}$$

not in the state z but in the vector $\varsigma^T = \left[|z_1|^{\frac{1}{2}} sign(z_1), z_2\right]$, with a particular symmetric an positive definite P matrix. The function in [20] is continuous and differentiable except on the set $\{(z_1, z_2): z_1 = 0\}$. Moreover, the solutions of the system (32) before arriving to the equilibrium point $z_1 = z_2 = 0$ every time crossing the line $z_1 = 0$ when $z_2 \neq 0$. This means that the derivative of the function [20] exist almost every time till the moment when $z_1 = z_2 = 0$. It was shown in [20] that function is a strong Lyapunov function for (32), *i.e.*, its derivative $\dot{V}$, when it exists, is negative definite when $K_1, K_2 > 0$.

It is very surprising that the construction of the suitable matrices $P = P^T > 0$ that provide a Lyapunov function can be reduced to the solution of an algebraic Lyapunov equation

$$A^T P + PA = -Q, \tag{34}$$

with a special selection for the A matrix. This nice and unexpected fact allows us to reduce the construction of Lyapunov functions for the highly nonlinear STA (32) to the usual linear procedure of solving the equation (34)

Proposition 2: Consider the matrix

$$A = \begin{bmatrix} \frac{1}{2}K_1 & \frac{1}{2} \\ -K_2 & 0 \end{bmatrix}, \tag{35}$$

where $K_1 > 0, K_2 > 0$, so that A is Hurwitz. For every symmetric and positive definite matrix $Q = Q^T > 0$, the function (33) is global, *strong Lyapunov function* for the STA (31), where $P = P^T > 0$ is the unique symmetric and positive definite solution of (34). Moreover, the derivative $\dot{V}$ of Lyapunov function satisfies the differential inequality almost every where

$$\dot{V} \leq -\gamma(Q)V^{\frac{1}{2}}(x), \tag{36}$$

where

$$\gamma(Q) \triangleq \frac{\lambda_{min}(Q)\lambda_{min}^{\frac{1}{2}}(P)}{\lambda_{max}(P)}, \tag{37}$$

is a scalar depending on the selection of the Q matrix.

Proof: A is Hurwitz, since its characteristic polynomial is $p(s) = s^2 + \frac{1}{2}K_1 s + \frac{1}{2}K_2$. The fact that under the given conditions the solution P of (34) satisfies the stated properties is standard in control [25]. Therefore, $V(z)$ in (33) is continuous, positive definite and radially unbounded in $\mathbb{R}^2$. Furthermore, it is differentiable everywhere, except on the line $\{z_1 = 0\}$. Since the trajectories of (32) cannot stay on this set, before reaching the origin, $\dot{V}$ can be calculated in the usual way. Since

$$\dot{\zeta} = \left[\frac{1}{2}\frac{1}{|x_1|^{\frac{1}{2}}}\left(-K_1|z_1|^{\frac{1}{2}}sign(z_1) + z_2\right) - K_2 sign(z_1)\right] = \frac{1}{|\zeta_1|}A\zeta, \tag{38}$$

it follows that

$$\dot{V} = \frac{1}{\zeta_1}\zeta^T(A^T P + PA)\zeta = -\frac{1}{|\zeta_1|}\zeta^T Q\zeta. \tag{39}$$

Now, from the standard inequality for quadratic forms

$$\lambda_{min}\{P\}\|\zeta\|_2^2 \leq \zeta P\zeta \leq \lambda_{max}\{P\}\|\zeta\|_2^2, \tag{40}$$

where $\|\zeta\|_2^2 = \zeta_1^2 + \zeta_2^2 = |z_1| + z_2^2$ is the Euclidean norm of ζ, and the inequality

$$|\zeta_1| = |z_1|^{\frac{1}{2}} \leq \|\zeta\|_2 \leq \frac{V^{\frac{1}{2}}(z)}{\lambda_{min}^{\frac{1}{2}}\{P\}}, \tag{41}$$

it follows easily that (36) is satisfied, so that $V(z)$ is a strong Lyapunov function.

ACKNOWLEDGMENT

The authors are grateful with the editor and the reviewers for their valuable comments and suggestions that allowed improvement significant of this work.

This research was supported by Secretaría de Investigación y Posgrado del Instituto Politécnico Nacional (SIP-IPN). Also, this work was partially supported by the Mexican CONACYT (Ciencia Básica) under Grant No. 105799, PAICYT UANL and Project PCP-CONACYT. Additionally, Jesús Martinez acknowledges the support received by CCA IPN and CONACYT for conducting his research stay at the UANL.

REFERENCES

[1] S. Chung, and J. Slotine, “Cooperative Robot Control and Concurrent Synchronization of Lagrangian Systems”, *IEEE Transactions on Robotics*, vol. 25 (3), pp. 686-700, 2008.

[2] S. Chung, “Nonlinear Control and Synchronization of Multiple Lagrangian Systems with Applications to Tethered Formation Flight Spacecraft”, Ph.D. Thesis. Massachusetts Institute of Technology (MIT), 2007.

[3] H. Nijmeijer, and A. Rodríguez, *Synchronization of Mechanical Systems*. Series Editor: Leon O. Chua. World Scientific Publishing Co. Pte. 2003.

[4] A. Rodríguez, and H. Nijmeijer, “Coordination of two robot manipulators based on position measurements only”, *Dynamics and Control Reporter*, vol. 23, 2000.

[5] V. Parra, S. Arimoto, Y. Liu, G.Hirzinger, and P. Akella, “Dynamic Sliding PID Control for Tracking of Robot Manipulators: Theory and Experiments”, *IEEE Transactions on Robotics and Automation*, vol. 19, no. 6, pp. 967-976, December 2003.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

[6] L. M. Capisani, A. Ferrara and L. Magnani, "Design and experimental validation of a second-order sliding-mode motion controller for robot manipulators", *International Journal of Control*, vol. 82/2, pp. 365-377, 2009.
[7] J. Davila, L. Fridman, and A. Levant, "Second-Order Sliding-Mode Observer for Mechanical Systems", *IEEE Transactions on Automatic Control*, vol. 50, no.11, pp. 187-193, November 2005.
[8] A. Levant, "Higher-order sliding modes, differentiation and output-feedback control", *International Journal of Control*, vol. 76, No. 9/10, pp. 924-941, 2003.
[9] Y. Shtessel, et al., *Sliding Mode Control and Observation*. Birkhauser, Springer Science+ Business Media New York, 2014.
[10] G. Bartolini, L. Fridman, et al., *Modern Sliding Mode Control Theory*. Springer-Verlag Berlin Heidelberg, 2008.
[11] W. Perruquetti, and J. Barbot, *Sliding Mode Control in Engineering.* Marcel Dekker, Inc., New York Basel, 2002.
[12] A. Pisano, E. Punta and E. Usai "A survey of applications of second-order sliding mode control to mechanical systems", *International Journal of Control*, vol. 76 No.9-10, pp. 875-892, 2003.
[13] V. Utkin, et al., *Sliding Mode Control in Electro-Mechanical Systems.* By Taylor Francis Group, Second Edition, United States of America, 2009.
[14] A. Dávila, J. A. Moreno, and L. Fridman, "Optimal Lyapunov function selection for reaching time estimation of Super Twisting Algorithm", joint in *48th IEEE Conference on Decision and Control and 28th Chinese Control Conference Shanghai*, P.R. China, December 16-18, 2009.
[15] C. Ma, et al., "Synchronization of networked Euler–Lagrange systems by sampled-data communication with time varying transmission delays under directed topology", *Neurocomputing*, 2014.
[16] A. Abdelkader, et al., "Synchronization of Lagrangian Systems with Irregular Communication Delays", *IEEE Transactions on Automatic Control*, vol. 59, No.1, pp. 187-193, January 2014.
[17] H. Sira-Ramírez, et al., *Control de Sistemas no Lineales*. CEA y Prentice-Hall, 2005.
[18] L. Derafaa, A. Benallegueb, A. and L. Fridman, "Super twisting control algorithm for the attitude tracking of a four rotors UAV", *Journal of the Franklin Institute*, Published by Elsevier Ltd., Vol. 349, pp 685-699, 2012.
[19] J. Rivera, et al., "Super-Twisting Sliding Mode in Motion Control Systems", *Sliding Mode Control, Prof. Andrzej Bartoszewicz (Ed.)*, ISBN: 978- 953-307-162-6, InTech, 2011.
[20] A. Levant, "Higher-order sliding modes, differentiation and output-feedback control", *International Journal of Control*, vol. 76, No. 9/10, pp 924-941, 2003.
[21] C. Edwards, et al., *Advances in Variable Structure and Sliding Mode Control*. LNCIS 334, pp 143-168, Springer-Verlag Berlin Heidelberg, 2005.
[22] A. F. Filippov, *Differential equations with discontinuous righthand side.* Kluwer. Dordrecht, The Netherlands. 304 p. 1988.
[23] A. Levant, and A. Michael, "Adjustment of high-order sliding-mode controllers", *International Journal of Robust and Nonlinear Control*, vol.19 no.15, pp.1657-1672. 2009.
[24] J. A. Moreno, and M. Osorio, "A Lyapunov approach to second-order sliding mode controllers and observers", in *47th IEEE Conference on Decision and Control*, CDC 2008, pp.2856-2861, 2008.
[25] H. K. Khalil, *Nonlinear Systems.* Third ed. Prentice-Hall. Upsaddle River, New Jersey. 750 p. 2002.
[26] J. Martínez, S. Rodríguez, J. de León, and P. Niño, "Sincronización maestro-esclavo en Sistemas Dinámicos Lineales y no Lineales" *AMDM 2016 Tercer Congreso Internacional Sobre Tecnologías Avanzadas de Mecatrónica, Diseño y Manufactura*, Colombia, 2016.
[27] H. Nijmeijer, A. Rodriguez, *Synchronization of Mechanical Systems.* World Scientific Series on Nonlinear Science Co. Pte. Ltd., Leon O. Chua Editor, University of California Berkeley, 2003.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 9, páginas 89 - 97.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Integración de una plataforma y una grúa como prototipo para rehabilitación de niños con PCI

Rodríguez-Méndez Jessica E., Ramos-Saavedra Kassandra G., Ambriz-Sandoval Karen A., Hernández-Oliva Noemi, Alejandre-Flores Marisol.
Instituto Politécnico Nacional, CECyT 2 Miguel Bernard, Av. Nueva Casa de la Moneda No.133, Colonia Lomas de Sotelo, Delegación Miguel Hidalgo, C.P. 11200, Tel. (55)5729 6000, ext. 67059, Ciudad de México.
(affh.wii@gmail.com,tony_megadeth@hotmail.com,kaas1998@hotmail.com,nhernandezo@ipn.mx, marisolalejandre_flores@hotmail.com)

***Resumen*— El presente trabajo tiene como principal objetivo la integración de una plataforma y una grúa ortopédica que sirvan de asistencia en la rehabilitación de marcha para pacientes con parálisis cerebral infantil (PCI). Esta propuesta está formada por medio del diseño de un prototipo de entrenamiento de marcha, el cual tiene como función principal asistir a los terapeutas en la rehabilitación de niños con este tipo de discapacidad, la cual se enfoca únicamente en problemas para caminar adecuadamente, es decir en pacientes que no tienen comprometidas más funciones motoras de su cuerpo o estas son mínimas. Este tipo de terapias aplicadas de forma temprana ayudan considerablemente en el proceso de aprendizaje de marcha (caminar) del niño, con lo cual se espera que el niño pueda desplazarse de un lugar a otro por sí mismo. La integración que se propone para el desarrollo del presente prototipo, está conformado por una estructura mecánica que consta de barandales de seguridad y una plataforma de marcha en una caminadora, así como de la adecuación de una grúa ortopédica con arnés para la sujeción de los niños, estas se implementan con la finalidad de levantar y movilizar a los pacientes que no puedan sostenerse totalmente por sí mismos, la cual es ajustada mediante el arnés y este va de acuerdo a la estatura del paciente. El control de la plataforma de marcha es realizada por medio de una tarjeta de control MC-60, con sus diferentes elementos eléctricos y electrónicos que la componen, como son botones pulsadores, un potenciómetro lineal, entre otros; los cuales permiten la variación de la velocidad de la caminadora y así poder someter al paciente a diferentes niveles de intensidad en la marcha. Estos grados de progresión son determinados mediante el avance de la terapia y la valoración del fisioterapeuta encargado. Una vez ensamblados todos los dispositivos y como parte de la integración final, se realizaron algunas adecuaciones con el fin de garantizar la seguridad del paciente en su rehabilitación así como otras propuesta a considerar a futuro, a la par con pruebas de validación del mismo para las operaciones de arranque y paro.**

I. INTRODUCION

Actualmente un problema de salud que se ha ido incrementando es el de casos con niños con alguna discapacidad, como lo es la parálisis cerebral (PCI), los cuales necesitan terapias físicas que les ayuden a caminar, moverse o alguna otra capacidad motriz. En ese aspecto en México, el Centro de Cirugía Especial a través de la Oficina de Representación para la Promoción e Integración Social para personas con Discapacidad de la Presidencia de la Republica, reporta como estadísticas que cada año aumentan alrededor de 270 mil personas con discapacidad en el país, de los cuales 12000 corresponden con niños con algún tipo de parálisis cerebral; sin embargo estas cifras únicamente son estimaciones ya que como tal no existe en el país un registro exacto de pacientes, médicos y centros especializados los cuales estén enfocados específicamente a llevar un control total de la población involucrada. Por otro lado, para la Organización Mundial de la Salud (OMS), calcula que el 14% de la población mexicana sufre de alguna discapacidad, estos datos han sido revisados en [1]. Con lo cual puede notarse que este problema no es exclusivo del país, es una afectación a nivel mundial, teniendo diversos factores que lo generan, sin embargo el objetivo de este trabajo no es centrarse en el diagnostico o causas que lo originan, la parte esencial radica en el hecho de conocer a grandes rasgos que es y algunos de los tipos de rehabilitación, por lo cual si se está interesado en conocer a detalle la enfermedad se recomienda revisar y consultar los tipos de PCI y sus afectaciones con mayor detalle en [2], [3] y [4], así como en la guía del Sector Salud para el diagnostico del niño con parálisis cerebral [5].

Retomando el interés en el tema, nos enfocaremos en los avances que se han obtenido en años recientes en el proceso de rehabilitación, es en este sentido que áreas como la mecatrónica ha permitido incrementar el desarrollo, mediante propuestas de integración y diseño de algunos prototipos que permitan a los pacientes con este tipo de discapacidad mejorar su calidad de vida, mediante el fortalecimiento de sus extremidades inferiores, de sus capacidades de relacionarse con el entorno, entre otras; todas con aportaciones específicas y con validaciones experimentales como con pacientes, obteniendo una mejora y una respuesta positiva en su uso, sin embargo aún hay mucho más por diseñar. Posteriormente se realizara un análisis detallado de algunos prototipos que se encontraron en la literatura y los cuales representan una valiosa aportación al prototipo que se desarrollara en el presente artículo, debido a que incluyen diferentes tipos de estructuras en su diseño, sin embargo, hasta hace poco tiempo no todos estos se enfocaban en niños o se requiere de dispositivos externos para lograr un proceso de rehabilitación, lo que dificulta el proceso mismo. En ese sentido la Red de Expertos en Robótica y Mecatrónica del Instituto Politécnico

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Nacional (RERYM IPN) se ha propuesto desarrollar el diseño de un prototipo que considere las características y necesidades de pacientes menores a fin de brindar un desarrollo propio con el cual se genere una opción más para dispositivos robóticos para rehabilitación en niños con PCI, en ese sentido el trabajo está dividido a través de grupos de trabajo denominados nodos, en el correspondiente con el CECyT 2 Miguel Bernard, esta aportación está enfocada al trabajo que se desarrolla con alumnos de nivel medio superior por medio de la integración de un prototipo de tipo experimental, el cual contempla de forma general como se activaría o manipularía el prototipo, es decir operaciones básicas de control como lo es el arranque y paro del mismo.

El presente artículo está organizado de la siguiente forma en la sección II se describe la integración de la estructura plataforma-grúa, incluyendo las consideraciones y el análisis de diferentes prototipos que existen hoy en día; en la sección III se presentan los elementos eléctricos y electrónicos así como la descripción de la selección de los dispositivos empleados. En la sección IV, se presenta la integración de los componentes y el ensamble final del prototipo. Las pruebas realizadas para validar el funcionamiento del prototipo se presentan en la sección V. Finalmente en la sección VI se presentan las conclusiones del artículo y posibles opciones de mejora en el diseño del control del prototipo.

II. INTEGRACIÓN DE LA ESTRUCTURA PLATAFORMA-GRÚA

Como ya se mencionó, el presente trabajo ha sido desarrollado con el fin de aportar los conocimientos adquiridos en el tema al resto de la comunidad de la RERYM IPN, por lo tanto y considerando las conocimientos básicos y a nivel técnico de los participantes, el interés principal se enfoca en realizar operaciones básicas de arranque y paro del prototipo por medio de adecuaciones eléctricas para en un futuro proponer el control formal para la operación de la propuesta final, para lograr el objetivo se retomaron e integraron componentes como lo es una caminadora a la cual se le agrego una grúa con arnés y un barandal con el fin de mostrar el funcionamiento de un prototipo como los que existen actualmente.

Pero para lograr dicho objetivo, se realizaron investigaciones de propuestas existentes, de las cuales se consideró como una de las más importantes la que se presenta en [7], puesto que en esta se muestra un análisis exhaustivo de diferentes sistemas robóticos utilizadas que se emplean en la rehabilitación funcional de miembro inferior, en este breve compendio se describen 50 dispositivos desarrollados hasta el año de su publicación en 2014, estos fueron clasificados en tres a) robots de aplicación clínica o prototipos, b) de tipo exoesqueleto y c) sistemas híbridos (sistemas de efector final); se recomienda leer detalladamente este articulo pues es la base principal de la cual se tuvo un concepto general de los diferentes sistemas que existen, ya que pueden ser fácilmente ubicados en tablas con las que se especifican detalles importantes como el nombre, desarrollador, tipo de terapia, posición del paciente, articulaciones o grados de libertad con que cuentan, tipo de robot, estado de desarrollo entre otros. Con este artículo se inició con la búsqueda en específico de algunos de los prototipos mostrados, ya que al ser un compendio esto facilito la elección final para la propuesta a presentarse mas adelante.

De este trabajo se decidió enfocarse en el análisis de algunos sistemas, especialmente en los que se describen y detallan a continuación; en [8,9], el proceso de rehabilitación por locomoción es realizado por medio de una ortesis robótica, esta se encuentra adaptada como un solo dispositivo de marcha, el cual denominaron como LOKOMAT, con este se obtienen los cálculos correspondientes para las terapias y se reportan las pruebas de simulación por medio de algoritmos automáticos de adaptación de patrones de marcha, con estos resultados validan la importancia del empleo de este tipo de terapias. Mientras que en [10], se trata acerca del exoesqueleto LOPES y lo que denominan como una relación viable de un dispositivo implementado en la marcha, estos resultados fueron reportados por medio de la realización de pruebas en rehabilitación, en las que se concluye que cumplen con la asistencia en la marcha. Por otra parte en [11], se muestra el diseño de otro tipo de exoesqueleto para la asistencia y entrenamiento de la marcha en terapias de rehabilitación de pacientes con deficiencias motoras, centrándose en el caso de lesiones sufridas después de un accidente cerebrovascular o de una lesión de la medula espinal, para ello se diseñaron pruebas de validación por medio del empleo de dos exoesqueletos GBO y Alex, el cual además es motorizado. Otro dispositivo de tipo exoesqueleto revisado corresponde con [12], el cual funciona cuando el paciente camina sobre un tapiz rodante el cual puede estar o no apoyado por medio de un sistema de suspensión, este fue probado por medio de experimentos clínicos, previendo varias consideraciones a futuro como un estudio con mayor número de pacientes. Otra referencia de gran importancia basada en mecanismos que aporten cierta ayuda mediante los procesos de marcha, pero sobre todo que cumplan con el análisis de pacientes normales contra algunos que han sufrido algún daño, a partir de este tipo de estudios se han podido obtener resultados favorables en pacientes que después de cuatro semanas de entrenamiento diario han recuperado parte de esta movilidad mediante el empleo de este tipo de prototipos, como puede describirse en [13].

Otro sistema desarrollado es GE-O [14], este permite que los pacientes a rehabilitar la marcha, este en particular fue diseñado como un sistema de efector final, el cual a partir de las trayectorias de las placas del pie para ser

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

programadas en el mismo, lo cual permite obtener simulaciones antes de la práctica incluyendo subir y bajar escaleras, para lograr estos patrones de activación muscular del miembro inferior de sujetos con Hemiparesia se coloca al paciente en el dispositivo y se verifica el avance en con la máquina. Gait trainer [15], son prototipos diseñados por Rifton con una variedad de desarrollos que permiten apoyar al paciente con alguna deficiencia neurológica, este distribuidor cuenta con diversos desarrollos enfocados desde el entrenamiento de la movilidad temprana hasta soportes para personas adultas, es importante mencionar que esta empresa también ha realizado investigación detallada de las aportaciones de los prototipos que distribuyen siendo de gran interés el articulo presentado en [16], donde se hace referencia a la importancia de la rehabilitación de la marcha en niños ya que esto les ayuda a mejorar su adaptación con su entorno y lo que denominan modelos terapéuticos, entre otros estudios más publicados por ellos mismos.

Como se ha mencionado con anterioridad existen gran cantidad de sistemas robóticos empleados como apoyo en la realización de terapias que permitan mejorar el proceso de aprendizaje de la marcha. En el caso del presente trabajo y con el fin de validar las opciones propuestas de arranque y paro de un sistema similar, se opto por combinar una estructura del tipo de LOKOMAT, la cual está enfocada en la marcha del paciente y su estructura esta compuesta por una banda de trote y una grúa con arnés, otro dispositivo similar es el sistema robótico GE-O y el Gait trainer anteriormente descritos.

Para ello se dividieron como dos componentes principales para la integración de la propuesta final, la primera corresponde con la plataforma de marcha (caminadora) con dos barandales de seguridad que fueron adaptados, y la segunda con una grúa y el arnés. Un primer bosquejo de la idea se muestra en la Fig. 1 correspondiente con el dibujo acotado de las vistas del prototipo, con acotaciones en centímetros.

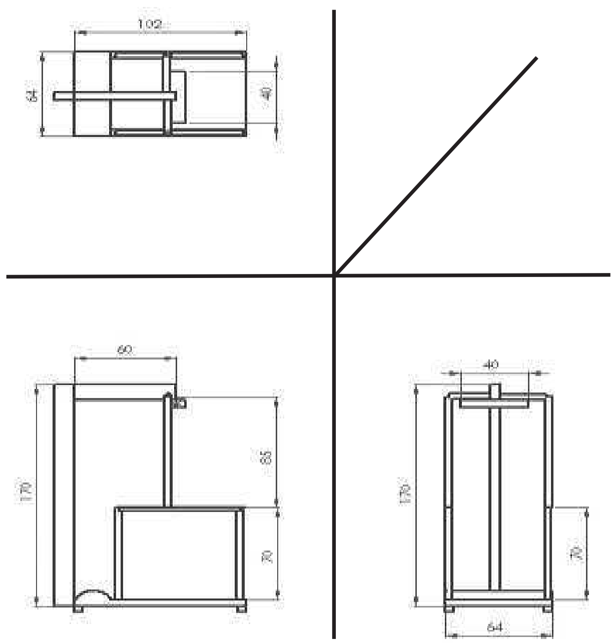


Figure 1. Dibujo en isometrico del prototipo propuesto (Acotaciones en cm).

A continuación se describirán con mayor detalle en qué consisten ambas estructuras, así como las consideraciones y dispositivos empleados en el diseño del prototipo.

A. Plataforma

La plataforma que se decidió emplear con el fin de validar las pruebas eléctricas de arranque y paro, corresponden con las de una caminadora que tiene integrada una tarjeta de control MC-60 y un potenciómetro lineal con el fin de regular la velocidad de la misma durante la puesta en marcha. A esta se le agregaron unos barandales de seguridad, con la consideración de que al tratarse de niños con PCI estos deberían tener un apoyo durante la terapia de marcha.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

B. Grúa ortopédica con arnés

Al emplear una caminadora de este tipo, se observó que era necesario adaptarle una grúa como las que se utilizan en ortopedia, en conjunto con un arnés de sujeción para el niño que recibirá la terapia física de rehabilitación, esta fue soldada y adecuada a la caminadora, la forma de operación es mediante poleas las cuales permiten ajustar el arnés de acuerdo con el tamaño del niño. Sin embargo es importante aclarar que en esta primer etapa, ha sido contemplado reutilizar los elementos con los que contaba la caminadora inicialmente, esto con el fin de que los alumnos involucrados se familiarizaran con el funcionamiento del motor y los rodillos en conjunto. Dando un mayor énfasis en la conformación de la estructura mecánica, por ello el diseño final en SolidWorks® se muestra en la Fig. 2.

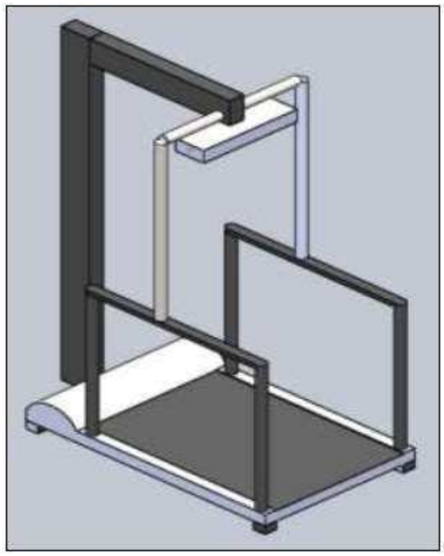


Figure 2. Prototipo diseñado.

III. Elementos eléctricos electrónicos de la plataforma de marcha

Como ya se ha mencionado, en cuanto a la estructura mecánica, y la adecuación de la grúa, arnés y barandales primero fue necesario seleccionar los componentes apropiados que permitan el funcionamiento de la caminadora.

La plataforma de marcha está integrada principalmente por una tarjeta de control MC-60 (Fig. 4) cuya función principal es la de controlar la cinta transportadora sobre la cual se colocara al paciente.

En esta primera etapa se decidió trabajar con los elementos propios de la caminadora con el fin de poder hacer las primeras pruebas de funcionamiento del prototipo.



Figure 3. Tarjeta de controlador MC-60.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

La tarjeta de control MC-60 del motor permite la regulación de la velocidad, puesta en marcha y paro de la caminadora por medio de los diferentes dispositivos electrónicos con los que cuenta.

La tarjeta está integrada por una sección donde se encuentra ubicada una fuente de alimentación regulada que va de los 12 VCD a los 120 VCA la cual tiene integrada un limitador de corriente cuya característica principal es la de limitar la cantidad de corriente a la carga de la caminadora para evitar daños bajo las pruebas realizadas.

También consta de un arranque suave que puede controlar la tensión del motor de forma que esta aumente gradualmente durante la puesta en marcha, esto significa que el motor se pone en marcha suavemente y que los esfuerzos mecánicos y eléctricos se reducen al mínimo y esto permite también una parada suave.

Tiene varios leds que se utilizan para señalización de algunas funciones, las principales que se emplearon y su correspondiente indicación se describirán de la siguiente forma:

- LED6 - es la activación en el controlador, si este no está activado, indica que no hay tensión que enviar al motor. Este LED puede variar la intensidad del brillo en función de la configuración de la velocidad. Si este, no se enciende cuando la llave de seguridad se inserta y el LED7 está encendido, el controlador tendrá que ser reemplazado.
- LED7 - indica que una señal de control de velocidad se está recibiendo desde la consola. Este led parpadea cuando se recibe una señal de velocidad. Si este no se enciende cuando la velocidad se fija por encima del valor de 0 km/h, debe encenderse el LED11, con lo cual se estaría indicando un problema con los cables o alambres de velocidad de la consola.
- LED11 - indica que el controlador está dando salida a 12 Vcc para el potenciómetro de velocidad. Si este no se enciende cuando se aplica voltaje, entonces de debe reemplazar el controlador ya que indica de una falla mayor.
- LED16 - "CUR LIM". Este se ilumina cuando el amperaje de la máquina para correr está cerca del límite de corriente fijado, indicando de una posible falla a los diferentes sistemas.

Algunos otros detalles de interés de la tarjeta son el potenciómetro de tipo lineal (Fig. 4) y los diferentes componentes que han sido empleados son descritos con mayor detalle en [17].



Figure 4. Potenciometro lineal empleado.

El potenciómetro lineal tiene las siguientes características técnicas: Rango desde 35mm hasta 950mm. Linealidad desde 0.2 hasta 0.075%. Salida: Resistiva de 0 a 10kΩ, este dispositivo nos permite variar la velocidad de la caminadora que va desde 0.05 km/h a 10km/h.

Por otra parte el motor empleado en la caminadora se muestra en la Fig. 5, este tiene como característica principales que al alimentarse con un voltaje máximo de $95V_{CD}$ corresponde con una velocidad de 10 km/h, mientras que con el voltaje mínimo $20V_{CD}$ la velocidad seria de 0.05km/h, esta es regulada por medio del potenciómetro acoplado.



Figure 5. Motor de corriente directa.

Este motor trabaja con una potencia de 1 HP (caballo de fuerza), lo cual le propicia que la banda transportadora y los rodillos giren una vez que desde la tarjeta de control se envíen las señales necesarias para su operación.

El encendido y apagado de la caminadora se realiza mediante un interruptor miniatura de tipo balancín como se muestra en la Fig. 6, cuyas características son de 4 Amperes, 250/127 VCA, de un polo y un tiro con dos posiciones (ON-OFF).



Figure 6. Interruptor miniatura

Finalmente en la Fig. 7 se muestran todos los componentes eléctricos que integran el prototipo.



Figure 7. Componentes del prototipo propuesto.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

En esta seccion se dieron a conocer los elementos eléctricos electrónicos de la caminadora asi como las caracteristicas tecnicas de los diferentes dispositivos que se utilizaron en la puesta en marcha del prototipo, para conocer las medidas de seguridad que se deben de tener en cuenta según los diferentes elementos eléctricos electrónicos que intervienen en el prototipo.

IV. Integración de los componentes y ensamble final

Una vez que fueron revisados y acoplados los componentes electrónicos y eléctricos, se propuso una grúa ortopédica y unos barandales adaptados a la caminadora, como ya se mostró anteriormente. La razón principal de esto se debió a que por seguridad del paciente en rehabilitación era necesario adecuarle unos barandales a lo largo de la caminadora con el fin de que pudiese tener un punto para sujetarse y con ello sentirse seguro.

Por otro lado inicialmente se había contemplado un sistema de poleas a las cuales se estaría adaptando un chaleco de seguridad, que tiene como función principal dar soporte en la espalda del paciente, este por medio de las poleas se ajustaría a la estatura del niño en terapia; sin embargo por el momento se ha implementado una estructura con un perfil tubular recto (PTR) de 2”x1 ½” (pulgada), de la base del prototipo se encuentra ensamblado verticalmente con una altura de 170cm, junto con un trozo formando un ángulo de 90° unido a la parte principal, que servirá para dar soporte al arnés, este cuenta con cuatro armellas de 5/16” que sostienen cada una un gancho de seguridad para arnés, de donde se sujeta el mismo chaleco de seguridad.

Por otro lado y considerando darle mayor soporte a la estructura, se colocó otra estructura de aluminio de 4cm de diámetro en forma de “Y invertida”, la cual fue soldada a los barandales que se encuentran a lo largo de la caminadora. En la Fig. 8 se muestra como se encuentra actualmente el prototipo ensamblado.



Figure 8. Prototipo ensamblado.

V. Pruebas realizadas

Al ser un proyecto desarrollado en pocos meses aún hay varias opciones de mejora y modificaciones que pueden realizarse a futuro, sin embargo una vez ensamblado el prototipo se realizaron varias pruebas de funcionamiento. Estas se limitan únicamente a verificar que la cinta transportadora de la caminadora esta funcionando por medio de las variaciones de voltaje que se tienen a través del potenciómetro lineal integrado con la tarjeta de control MC-60;

en esta fase inicial se conto con el consentimiento oral de los padres, puesto que las pruebas realizadas son de riesgo mínimo y no contemplan la necesidad de toma de muestras biológicas, sin embargo existe confidencialidad con respecto al uso y difusión de imágenes de ambos niños que participaron. Es decir, las pruebas se encuentran divididas en el caso de dos niños, estas consisten en colocar al niño sobre la plataforma, asumiendo que estos no tienen problemas con la marcha que realizan, el objetivo únicamente es comprobar si existe una comunicación adecuada de los diferentes dispositivos, para ello se tienen dos sujetos de prueba, el primero es el sujeto A (niña de 5 años y una estatura de 112cm) se ajusto la velocidad de la caminadora seleccionando el nivel más bajo de intensidad, en este caso la velocidad empleada fue de 0.5km/h el cual es equivalente con una tensión eléctrica de $20V_{CD}$. La prueba 2 fue realizada con el sujeto B (niño de 12 años y una estatura de 165cm), al cual se le ajusto una velocidad mucho más rápida que la del sujeto A, esta corresponde con aproximadamente una tensión eléctrica de alrededor de 90 V_{CD}.

Es importante recalcar que estas pruebas corresponden con la primera parte del proyecto final, por lo que los niños son clínicamente sanos y su desempeño durante la marcha se considera normal para un niño con esas características, sin embargo y de acuerdo a las regulaciones existentes como es el caso de pruebas en niños con PCI, se requieren de un consentimiento informado de los padres, así como de uso y difusión de videos o imágenes con autorización de los mismos, el cual es avalado por la Norma oficial Mexicana (NOM), según reporta en Instituto Nacional de Salud Pública en su sitio de internet (INSP), como puede revisarse con mayor detalle en [18]. Para este caso de estudio los padres únicamente aprobaron la participación de los niños en la marcha sobre el prototipo pero no de la difusión de las imágenes, motivo por el cual no se presentan.

VI. CONCLUSIÓN

Una vez concluida la primera etapa del proyecto, en la cual únicamente se contempla el ensamblaje y puesta en marcha del prototipo, puede considerarse que resulto exitoso en su primera fase, ya que se cumplieron con las metas propuestas para este trabajo. Estas metas que han sido cubiertas totalmente corresponden con la fase de recopilación de información y búsqueda de dispositivos desarrollados (prototipos o patentes) para rehabilitación de marcha en pacientes con parálisis cerebral, los cuales por ahora no son diseñados específicamente para niños y solo en algunos casos tienen aditamentos especiales extra para este tipo de pacientes. A partir de este análisis detallado se optó por emplear la caminadora en conjunto con la grúa, esto debido en gran parte a la fácil manufactura y a una posible modificación del mismo, dando cierta libertad de cambiar la misma o alguno de los componentes eléctricos y electrónicos que la conforman. En ese sentido la función inicial del proyecto corresponde con la selección y adecuación de componentes eléctricos y/o electrónicos para controlar operaciones básicas como lo es el arranque y paro de un prototipo de rehabilitación para niños con PCI, en este aspecto y aprovechando la ventaja de tener la caminadora con la cinta de trote ajustada, se reutilizo la tarjeta de control de la misma y se le agrego un potenciómetro lineal que permitiera la variación de la velocidad de la cinta. Además se tiene un interruptor miniatura que enciende cuando el prototipo está trabajando, con lo cual se indica el arranque y puesta en marcha.

Sin embargo, el desarrollo de este se sigue trabajando con los alumnos integrantes del presente proyecto. Por lo que a futuro se espera modificar la etapa de control, mediante la implementación de un controlador programable y su etapa de potencia diseñada para esta aplicación, con lo que se espera que al existir fallas no se requiera el reemplazo completo de la tarjeta y con eso se simplifiquen la cantidad de componentes empleados en el prototipo actual.

Otra aportación significativa que puede ser mejorada consiste en la pintura de la misma, puesto que después de las pruebas con niños, se planteo la necesidad de tener colores mucho más adecuados y agradables. Otra mejora es en la estructura del arnés el cual puede ser mejorado basándonos en la información de los prototipos funcionales que existen en el mercado.

AGRADECIMIENTOS

Los autores agradecen al Instituto Politécnico Nacional a través de la Secretaria de Investigación y Posgrado por la aprobación del proyecto SIP20161413 y las facilidades para el desarrollo del mismo.

REFERENCIAS

[1] Ccem.org.mx. (2016). Estadísticas de Discapacidad en México. [Online] Available at: http://www.ccem.org.mx/statmex [Accessed 15 Oct. 2016]

[2] Póo Argüelles, P. (2016). Parálisis cerebral infantil. Protocolos Diagnóstico Terapéuticos de la AEP: Neurología Pediátrica, pp.271-277.

[3] Damiano, D., Alter, K. and Chambers, H. (2009). New Clinical and Research Trends in Lower Extremity Management for Ambulatory Children with Cerebral Palsy. Physical Medicine and Rehabilitation Clinics of North America, 20(3), pp.469-491.

[4] http://www.cpdiscovery.com/ Cpdiscovery.com. (2016). Cerebral Palsy Discovery. [Online] Available at: http://www.cpdiscovery.com/ [Accessed 22 Oct. 2016]

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

[5] Martín Iglesias, A. and Collado Vázquez, S. (2015). Case in pediatric patients with neurological diseases: infant cerebral palsy. FisioGlía, 2(1), pp.5-8.
[6] http://www.cenetec.salud.gob.mx/. (2009). Resumen de Evidencias y Recomendaciones Evaluación diagnóstica del niño con parálisis cerebral en el tercer nivel de atención GPC Guía de Práctica Clínica Catálogo Maestro: DIF-332-09. [Online] Available at: http://www.cenetec.salud.gob.mx/descargas/gpc/CatalogoMaestro/332_DIF_09_Paralisis_cerebral/EyR_DIF_332_09.pdf [Accessed 21 Oct. 2016].
[7] Calderón-Bernal, A., Cano-de la Cuerda, R., Alguacil-Diego, I., Molina-Rueda, F., Cuesta-Gómez, A. and Miangolarra-Page, J. (2015). Terapia robótica para la rehabilitación de la marcha en patología neurológica. Rehabilitación, 49(3), pp.177-192.
[8] S. Jezernik, "Automatic Gait-Pattern Adaptation Algorithms for Rehabilitation with a 4-DOF Robotic Orthosis". IEEE transactions on robotics and automation, vol. 20, pp. 574-582, 2003.
[9] Manufacturer of the Lokomat, Pro, Technical Data LOKOMAT Pro, System training, pp. 2.
[10] J. Veneman, Design and Evaluation of the LOPES Exoskeleton Robot for Interactive Gait Rehabilitation. IEEE Transactions on neural systems and rehabilitation engineering, vol. 15, pp. 379-386, 2007.
[11] S. Agrawal, "Exoskeletons for Gait Assistance and Training of the Motor-Impaired". IEEE 10th International Conference on Rehabilitation Robotics, pp. 1108-1113, 2007.
[12] S. Hesse, "A mechanized gait trainer for restoration of gait". Journal of Rehabilitation Research and Development, vol. 37, pp. 701-708, 2000.
[13] N. Koceska, S. Koceski, P. Beomonte Zobel and F. Durante, Chapter 11. Gait Training-using Pneumatically Actuated Robot System. En Alejandra Barrera (Ed), InTech. Advances in Robot Navigation, pp. 223 – 238, Julio 2011.
[14] Hesse, S., Waldner, A. and Tomelleri, C. (2010). Innovative gait robot for the repetitive practice of floor walking and stair climbing up and down in stroke patients. Journal of Neuro Engineering and Rehabilitation, 7(1), p.30.)
[15] Rifton.com. (2016). Rifton | Resources and articles on the Rifton Pacer gait trainer. [Online] Available at: http://www.rifton.com/products/gait-trainers/pacer-gait-trainers?tab=resources [Accessed 18 Oct. 2016]
[16] Rifton.com. (2016). Rifton | Benefits of Early Mobility with an Emphasis on Gait Training. [Online] Available at: http://www.rifton.com/resources/articles/2011/february/benefits-of-early-mobility-with-an-emphasis-on-gait-training [Accessed 19 Oct. 2016]
[17] K. A. Ambriz-Sandoval, K. G. Ramos-Saavedra and J. E. Rodríguez Méndez, "Diseño de un control para la operación de sistemas mecatrónicos". Proyecto de titulación para la especialidad de Técnico en Maquinas con sistemas automatizados, Instituto Politécnico Nacional CECyT 2 Miguel Bernard, Junio 2016.
[18] Insp.mx. (2016). Consentimiento Informado. [Online] Available at: https://www.insp.mx/insp-cei/consentimiento-informado.html [Accessed 13 Oct. 2016].

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 10, páginas 98 - 110.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Control por par calculado para un seguidor solar de dos grados de libertad

Sergio Isai Palomino-Resendiz[1], Diego Alonso Flores Hernández[2], Alberto Luviano Juárez[3], Norma Lozada Castillo[4] e Isaac Chairez[5]

Resumen— En el presente trabajo se realiza el modelado dinámico y control de un Robot Seguidor Solar de dos grados de libertad en tareas de seguimiento de trayectoria. El cálculo de la trayectoria deseada se basa en el uso de la ecuación de Cooper. El entorno de simulación propuesto considera el uso de las propiedades físicas del material mediante la mezcla de un software de diseño (SolidWorks) y otro de simulación numérica (Matlab-Simulink). Se utiliza un controlador de tipo par calculado, el cual realiza la tarea de seguimiento con buena precisión.

I. Introducción

Actualmente el uso y producción de energía limpia (solar, eólica, entre otras) presenta una solución favorable a la crisis energética, sustentabilidad e impacto ambiental que el mundo vive, resultado de aplicar técnicas convencionales que procesan y transforman combustibles fósiles [1]. La energía solar como recurso, es una fuente inagotable útil para producir energía eléctrica (EE) a través de sistemas de conversión que implementan mecanismos e instrumentación para orientar dispositivos de captación (paneles fotovoltaicos (PV)) en relación a dos ejes (elevación (H) y azimutal (Z)) o grados de libertad (GDL) de forma perpendicular al Sol, en configuraciones estáticas o móviles (seguidor solar (SS)) [2], mejor conocidas como:

- Fija: Orientación manual de PV que permanece estática. Su máxima producción de EE sucede con incidencia solar directa o en el momento de mayor irradiancia del día.

- SS de un eje móvil (1GDL): Orientación manual para H o Z y mediante un algoritmo de búsqueda o posicionamiento rota el eje restante respecto a la trayectoria solar (TS). Presenta hasta un 30 por ciento de producción mayor de EE en comparación con el fijo para igual número de PV [7].

- SS de dos ejes móviles (2GDL): Orientación automática para ambos ejes mediante un algoritmo de búsqueda o posicionamiento en relación a la TS. Presenta hasta un 40 por ciento de producción mayor de EE en comparación con el fijo para igual número de PV [7].

- SS Híbrido: Configuraciones especializadas que utilizan más ejes e instrumentación. Su producción es igual o menor al SS de 2GDL, dado que el consumo energético es mayor.

El diseño y operación de este tipo de sistemas depende principalmente de la incidencia e irradiación solar, el número de PV, la calidad de los materiales y el consumo energético asociado al funcionamiento del mismo [3], todo con el objetivo de presentar la mayor cantidad de EE disponible. Convencionalmente la planeación o generación de trayectorias (ángulos asociados a cada eje) que orientan los PV de sistemas móviles, se obtienen de forma activa por el cálculo de la declinación solar (DS) y/o el uso de sensores (luminosidad, irradiancia solar, potencial eléctrico) implementados en métodos numéricos para micro-controladores o algoritmos computacionales y de forma pasiva o analógica mediante globos que contienen ciertos gases que reaccionan a la irradiancia solar atados a los PV, aunque este último método es muy poco utilizado por las

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

especializadas y costosas condiciones de operación que involucra [4]. La DS aproxima una TS fija sin considerar perturbaciones ambientales (nubes, sombras). Sin embargo, el costo computacional y económico es mucho menor en comparación con el consumo energético y los recursos necesarios para procesar, en tiempo real, el cálculo de una TS a partir de sensores, que dependiendo su calidad, calibración y el algoritmo de búsqueda pueden guiar al sistema de manera equivocada o redundante. La comparación en línea de ambos métodos permite obtener una TS más fiable, sin embargo, el costo económico y tiempo de procesamiento incrementan de manera considerable.

Adicionalmente, los algoritmos de búsqueda o posicionamiento pueden verse como un problema de seguimiento de trayectorias para los mecanismos del sistema (a través de sus actuadores), visto éste como un robot manipulador que por medio de técnicas clásicas como el control Proporcional Integral Derivativo (PID) y sus variantes, control por modos deslizantes, control por par calculado, entre otros, sean reproducidas [5]. Normalmente se eligen actuadores (motores de DC) con lector de posición (Encoder) y mecanismos reductores de engrane que bloquean su rotación para lapsos donde la actualización de posición no es significativa con el objetivo de reducir el consumo energético o rechazo físico de perturbaciones (condiciones del medio). Lo que resulta en un modelado dinámico más complejo del actuador y que para técnicas como el control PID y sus variantes o el control por modos deslizantes, significa compensar perturbaciones asociadas al funcionamiento y dinámica del sistema más el par mecánico (torque) de arranque de cada motor para cada movimiento constantemente, acción que implica un gran consumo energético en comparación con la técnica de control por par calculado que estima el Par necesario y suficiente para cada actuador en el momento que lo requiere considerando dichas condiciones de operación.

En relación al trabajo de Sidek [7], se reportó el estudio e implementación de un controlador PID basado en el seguimiento de TS generada por el cálculo de DS con datos obtenidos de un GPS y procesados por un micro-controlador PIC18f4431 para un SS de 2GDL. Básicamente la estrategia de solución calcula los valores de H y Z, luego realiza una comparación con el valor del Encoder de cada motor y actualiza la posición, el trabajo reporta una producción de 27 por ciento de EE mayor que un panel fijo. Sin embargo, no se considera el sistema de bloqueo por lo que en el lapso de actualización de posición el sistema se mantiene energizado, reduciendo la cantidad reportada de EE disponible real. Por otra parte, Huang [8] propuso un método particular de orientación de PV para un SS de 1GDL; estableciendo el ángulo del eje H para cada mes mediante 3 rotaciones de Z en el transcurso del día relacionadas a una medición de irradiancia solar que determina la posición del Sol al amanecer, el medio día y atardecer. Lo anterior incluye un consumo energético reducido relacionado al actuador por sistema de bloqueo, el cual se energiza solamente para actualizar la posición. El trabajo reportó hasta 38 por ciento de producción mayor de EE en comparación con un panel fijo, siendo una opción viable. Sin embargo, la propuesta no considera perturbaciones que pueden afectar por mucho tiempo una posición establecida a diferencia de una que puede actualizarse y tener un mejor desempeño en términos de producción, sin mencionar la calibración constante necesaria del eje H.

Adicionalmente, Fathabadi y Hassan [9] reportaron él diseño, construcción e implementación de un sistema SS de 2GDL para PV mediante DS, el cual obtiene los ángulos para H y Z, que actualizan la posición de un par de motores a pasos; el trabajo reporta hasta un 30 por ciento de EE disponible final considerando el consumo del sistema. A pesar de contar con mecanismos de engranes asociados a cada eje, el uso de motores a pasos involucra un consumo energético innecesario para mantenerlos en su posición además de que cada movimiento no es exacto sino aproximado en relación tamaño de paso.

La mayoría de las propuestas anteriores no consideran la dinámica del sistema mecánico. Es por ello que, en esta propuesta, se reducirá el consumo energético en el sistema de seguimiento mediante la implementación de un controlador por par calculado, que asegure el seguimiento de trayectorias resultantes del cálculo de la declinación solar sin realizar acciones de compensación de dinámicas no modeladas. Adicionalmente, se considerarán actuadores con mecanismos

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

reductores de engrane, logrando mediante los bloqueos de posición un menor consumo en los actuadores, lo que resultara en una cantidad de EE mayor disponible utilizando la configuración con mayor producción.

El resto del trabajo se presenta de la siguiente manera: la Sección II proporciona una descripción y desarrollo necesarios para el modelado de la propuesta de un seguidor solar de dos grados de libertad, además del desarrollo e implementación de un control por par calculado que asegure el posicionamiento correcto de los ejes siguiendo la trayectoria generada por el método numérico del cálculo de la declinación solar. En la Sección III se presentan las simulaciones numéricas mediante el uso de SolidWorks y Matlab a través del entorno de simulación Simulink-SimMechanics y, finalmente, se establecen conclusiones y propuestas de trabajo a futuro.

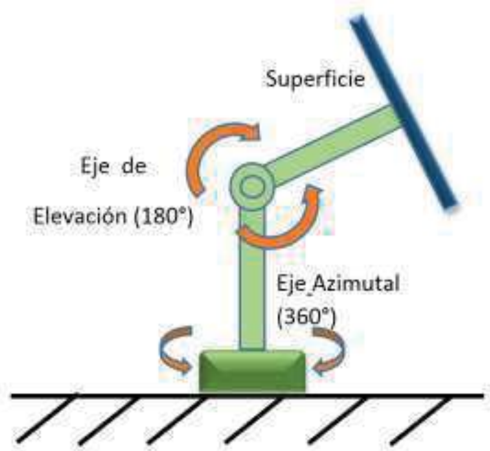


Fig. 1: Seguidor Solar de dos grados de libertad.

II. Preliminares

En esta sección de presentarán las herramientas necesarias para el desarrollo del presente trabajo en el siguiente orden: modelo dinámico, control y generación de trayectorias.

Notaciones

Para este trabajo se considerará $\mathbb{R}^{n\times m}$ como el espacio de matrices reales de n renglones y m columnas. Se adoptará la siguiente notación $c_i \coloneqq \cos(q_i)$, $s_i \coloneqq \operatorname{sen}(q_i)$ respectivamente.

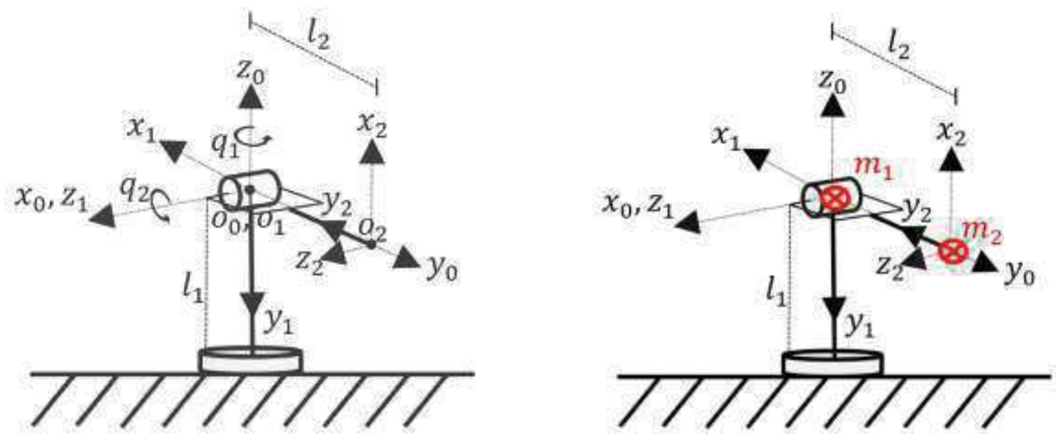
II-A. Modelado dinámico

Para obtener el modelo dinámico del robot ([10], [11], [12]), es necesario calcular primero el modelo cinemático directo por la metodología de Denavit-Hartenberg (D-H) y el Jacobiano asociado a cada centro de masa, dado el enésimo eslabón, J_n denotado por su respectivo Jacobiano geométrico.

Parámetros de D-H: La configuración del robot es revoluta- revoluta (RR) en cadena cinemática abierta; por las reglas de D-H se propone la localización de marcos coordenados y centros de

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

masa para cada grado de libertad además de un marco de referencia o tierra en un modelo equivalente descritos en la figura 2. Dónde: $O_0x_0y_0z_0$ es el sistema de referencia inercial del modelo, $O_ix_iy_iz_i$ los marcos coordenados, mi masa del eslabón, q_i ángulo de junta, l_i longitud respectiva de cada eslabón para $i = 1{,}2$, siendo i el eslabón asociado.



(a) (b)

Fig. 2: A la izquierda, se muestran los marcos coordenados obtenidos por la convención de D-H. A la derecha, se hace hincapié de los centros de masas.

Por convención, $O_0 = [0\ 0\ 0]^T$ y $z_0 = [0\ 0\ 1]^T$. Se tomarán l_1 como la distancia del marco inercial al centro de masa del eslabón 1, así como l_2 como la distancia del marco $O_1x_1y_1z_1$ al centro de masa del eslabón 2. Cabe mencionar que la distancia l_1 dada la propuesta de localización de los marcos coordenados y centros de masa puede ser depreciable sin afectar el desarrollo del cálculo. Así siguiendo la metodología de cinemática directa de D-H, se obtienen los siguientes parámetros (ver tabla 1):

i	a_i	α_i	d_i	θ_i
1	0	$\pi/2$	0	$\pi/2 + q_1$
2	l_2	0	0	q_2

Tabla 1. Parámetros de Denavit Hartenberg del seguidor solar

Así, la matriz de transformación homogénea que relaciona el marco del efector final $O_2x_2y_2z_2$ respecto al marco inercial está dada por

$$T_0^2 = A_1A_2$$

$$= \begin{bmatrix} -c_2s_1 & s_1s_2 & c_1 & -l_2c_2s_1 \\ c_1c_2 & -c_1s_2 & s_1 & l_2c_1c_2 \\ s_2 & c_2 & 0 & l_2s_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (1)$$

Con

$$A_1 = \begin{bmatrix} -s_1 & 0 & c_1 & 0 \\ c_1 & 0 & s_1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad A_2 = \begin{bmatrix} c_2 & -s_2 & 0 & l_2c_2 \\ s_2 & c_2 & 0 & l_2s_2 \\ 0 & c_2 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Jacobianos geométricos: Se obtiene el Jacobiano geométrico asociado al centro de masa del n-ésimo eslabón (con posición O_{cmn}), denotado por $J_n\ \mathbb{R}^{6\times 2}$ y sus respectivas divisiones en velocidad lineal y angular, descritas como $J_{vn}\ \mathbb{R}^{3\times 2}$ y $J_{wn}\ \mathbb{R}^{3\times 2}$, se tiene para $n = 1{,}2$.

$$J_n = \begin{bmatrix} J_{vn} \\ J_{wn} \end{bmatrix} \tag{2}$$

Esto es

$$J_1 = \begin{bmatrix} J_{v1} \\ J_{w1} \end{bmatrix} = \begin{bmatrix} z_0 \times (O_{cm1} - O_0) & 0 \\ z_0 & 0 \end{bmatrix}$$
$$J_2 = \begin{bmatrix} J_{v2} \\ J_{w2} \end{bmatrix} = \begin{bmatrix} z_0 \times (O_{cm2} - O_0) & z_2 \times (O_{cm2} - O_0) \\ z_0 & z_1 \end{bmatrix}$$

Usando A_1, T_0^2 y de la geometría de la figura 2, se tiene $O_{cm1} = O_1 = O_0$, $z_0 = [c_1\ s_1\ 0]^T$ y $O_{cm2} = [l_2c_2\ \ l_2s_2\ \ 0]^T$. Calculando entonces lo anterior:

$$J_{v1} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \qquad J_{w1} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{bmatrix} \tag{3}$$

$$J_{v2} = \begin{bmatrix} -l_2c_1s_2 & l_2s_1s_2 \\ -l_2s_1c_2 & -l_2c_1s_2 \\ 0 & l_2c_2 \end{bmatrix}, \qquad J_{w2} = \begin{bmatrix} 0 & c_1 \\ 0 & s_1 \\ 1 & 0 \end{bmatrix} \tag{4}$$

Luego

$$J_1 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{bmatrix}, \quad J_2 = \begin{bmatrix} -l_2c_1s_2 & l_2s_1s_2 \\ -l_2s_1c_2 & -l_2c_1s_2 \\ 0 & l_2c_2 \\ 0 & c_1 \\ 0 & s_1 \\ 1 & 0 \end{bmatrix}$$

Modelo dinámico general: Por convención [11], el modelo dinámico general tiene la forma:

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = u \tag{5}$$

Donde

- $q, \dot{q}$ variables y velocidades de junta, respectivamente.

- $D(q)\ \epsilon\ \mathbb{R}^{2\times 2}$ representa a la matriz de inercia, la cual es simétrica y definida positiva.

$$D(q) = \sum_{i=1}^{2} m_i J_{vi}^T J_{vi} + J_{wi}^T R_i I_i R_i^T J_{wi} \tag{6}$$

- $R_i\ \epsilon\ \mathbb{R}^{3\times 3}$, es la matriz de rotación $A_1, (i = 1), T_0^2 = A_1A_2 (i = 2)$ respectivamente.

- $I_i\ \epsilon\ \mathbb{R}^{3\times 3}$, es la matriz de componentes inerciales. Por hipótesis, para nuestro caso se define como una matriz diagonal, tal que la simetría y dinámica del mecanismo sea ideal. Al final dichas componentes de la matriz no contempladas serán compensadas por el controlador, vistas como una

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

perturbación.

$$I_i = \begin{bmatrix} I_{xi} & 0 & 0 \\ 0 & I_{yi} & 0 \\ 0 & 0 & I_{zi} \end{bmatrix}$$

- $C(q,\dot{q}) \epsilon \mathbb{R}^{2\times 2}$ es la matriz de fuerzas Centrípetas y de Coriolis cuyos elementos están dados por (7), con C_{ijk}, los llamados símbolos de Christoffel.

$$C(k,j) = \sum_{i=1}^{2} C_{ijk}\dot{q}_i \tag{7}$$

- $G(q) \epsilon \mathbb{R}^2$ es el vector de fuerzas de gravedad o pares gravitacionales obtenido como el gradiente de la energía potencial. La i-ésima componente de $G(q)$ está dada por:

$$G_i(q) = \frac{\partial P}{\partial q_i} \tag{8}$$

- $u \epsilon \mathbb{R}^2$ es el vector de par o torque de entrada del sistema su propuesta es generalizada.

$$u = [u_1\, u_2]^T \tag{9}$$

Para el caso de interés, de *(3)* y *(4)*, la matriz de inercia tiene la forma:

$$D(q) = \begin{bmatrix} cos^2(q_2)\left(m_2 l_2^2 + I_{y2}\right) + I_{y2}sen^2(q_2) + I_{y1} & 0 \\ 0 & m_2 l_2^2 + I_{2z} \end{bmatrix}$$

Los símbolos de Christoffel, elementos de la matriz de Coriolis $C(q,\dot{q})$, se definen a partir de los elementos de la matriz de inercia $D(q)$. Esto es:

$$C_{ijk} = \frac{1}{2}\{\frac{\partial d_{kj}}{\partial q_i} + \frac{\partial d_{ki}}{\partial q_j} - \frac{\partial d_{ij}}{\partial q_k}\} \tag{10}$$

Luego

$$C_{111} = \frac{1}{2}\left\{\frac{\partial d_{11}}{\partial q_1} + \frac{\partial d_{11}}{\partial q_1} - \frac{\partial d_{11}}{\partial q_1}\right\} = \frac{1}{2}\left\{\frac{\partial d_{11}}{\partial q_1}\right\} = 0$$

$$C_{112} = \frac{1}{2}\left\{\frac{\partial d_{21}}{\partial q_1} + \frac{\partial d_{21}}{\partial q_1} - \frac{\partial d_{11}}{\partial q_2}\right\} = \frac{1}{2}\left\{-\frac{\partial d_{11}}{\partial q_2}\right\}$$

$$C_{121} = \frac{1}{2}\left\{\frac{\partial d_{12}}{\partial q_1} + \frac{\partial d_{11}}{\partial q_2} - \frac{\partial d_{12}}{\partial q_1}\right\} = \frac{1}{2}\left\{\frac{\partial d_{11}}{\partial q_2}\right\} = C_{211}$$

$$C_{122} = \frac{1}{2}\left\{\frac{\partial d_{22}}{\partial q_1} + \frac{\partial d_{21}}{\partial q_2} - \frac{\partial d_{12}}{\partial q_2}\right\} = 0 = C_{212}$$

$$C_{211} = \frac{1}{2}\left\{\frac{\partial d_{11}}{\partial q_2} + \frac{\partial d_{12}}{\partial q_1} - \frac{\partial d_{21}}{\partial q_1}\right\} = \frac{1}{2}\left\{\frac{\partial d_{11}}{\partial q_2}\right\} = C_{121}$$

$$C_{212} = \frac{1}{2}\left\{\frac{\partial d_{21}}{\partial q_2} + \frac{\partial d_{22}}{\partial q_1} - \frac{\partial d_{21}}{\partial q_2}\right\} = 0 = C_{122}$$

$$C_{221} = \frac{1}{2}\left\{\frac{\partial d_{12}}{\partial q_2} + \frac{\partial d_{12}}{\partial q_2} - \frac{\partial d_{22}}{\partial q_1}\right\} = 0$$

$$C_{222} = \frac{1}{2}\left\{\frac{\partial d_{22}}{\partial q_2} + \frac{\partial d_{22}}{\partial q_2} - \frac{\partial d_{22}}{\partial q_2}\right\} = 0$$

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Con lo anterior, la matriz de Coriolis *(7)* es:

$$C(q,\dot{q}) = \begin{bmatrix} C(1,1) & C(1,2) \\ C(2,1) & C(2,2) \end{bmatrix}$$

Donde

$$C(1,1) = C_{111}\dot{q}_1 + C_{211}\dot{q}_2 = \cos(q_2)\,\mathrm{sen}(q_2)\,(m_2 l_2^2 + I_{y2} - I_{2x})\dot{q}_2$$
$$C(1,2) = C_{121}\dot{q}_1 + C_{221}\dot{q}_2 = -\frac{1}{2}\mathrm{sen}(2q_2)\,(-m_2 l_2^2 - I_{y2} + I_{2x})\dot{q}_1$$
$$C(2,1) = C_{112}\dot{q}_1 + C_{212}\dot{q}_2 = \cos(q_2)\,\mathrm{sen}(q_2)\,(m_2 l_2^2 + I_{y2} - I_{2x})\dot{q}_1$$
$$C(2,2) = C_{122}\dot{q}_1 + C_{222}\dot{q}_2 = 0$$

Por otra parte, para el vector de $G(q)$ de fuerzas gravitacionales *(8),* se considera la energía potencial

$$P = \sum_{i=1}^{2} m_i \vec{g} r_{ci}$$

Con $\vec{g} = [0\ 0\ g]^T$, $r_{ci} = O_i$, para $i = 1{,}2$. Así

$$G(q) = \begin{bmatrix} 0 \\ m_2 g h_2 \cos(q_2) \end{bmatrix}$$

II-B. Control por par calculado

El control por par calculado (CTC Computed Torque Control por sus siglas en inglés) es un controlador en lazo cerrado que es no lineal en las variables de estado respecto a un sistema asociado. La mayor ventaja del enfoque es que presenta estabilidad global así como una linealización exacta por realimentación del sistema, cuando éste es no lineal [11], [13]. Dada una trayectoria deseada q^* para el modelo dinámico *(5),* el control u por par calculado en su forma general es:

$$u = M(q)\ddot{q} + C(q,\dot{q})\dot{q} + +B(\dot{q}) + G(q) \qquad (11)$$

Donde

- $q, \dot{q}$ variables y velocidades de junta, respectivamente.
- $M(q) \in \mathbb{R}^{2\times 2}$ es la matriz de inercia más la inercia efectiva del motor, el cual está gobernado por la siguiente ecuación
$$M(q)\ddot{q} = D(q)\ddot{q} + \mathcal{J}, \quad \mathcal{J} = diag\{\mathcal{J}_1\ \mathcal{J}_2\} \qquad (12)$$

  $\mathcal{J}_1, \mathcal{J}_2$ son las inercias de los actuadores (motores DC) asociados a cada junta.
- $B(\dot{q})$ es la matriz de fricción seca y viscosa.

La propuesta de ley *(11)* supone que dada una posición y velocidad del actuador, se cancelen todas las no linealidades y por tanto, se tendrá el par necesario para aplicarlo a la inercia del actuador. Sustituyendo el control *(11)* en el modelo dinámico general *(5)* se tiene

$$M(q)\ddot{q} = M(q)\ddot{q}^*$$

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 10, páginas 98 - 110.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Por sustitución se tiene que $M(q)$ es definida positiva, y por tanto, es invertible; luego $\ddot{q} = \ddot{q}^{*}$. La relación anterior consiste en 2 controles de cadenas de integración de segundo orden, las cuales se pueden resolver mediante controles de tipo *PD* prealimentado.

Luego se tiene

$$u = M(q)[\ddot{q}^{*} - K_D(\dot{q} - \dot{q}^{*}) - K_P(q - q^{*})] + C(q, \dot{q})\dot{q} + G(q) \qquad K_D, K_P \in \mathbb{R}^{2\times 2}$$

Pictóricamente el lazo de control puede verse en la figura 3.

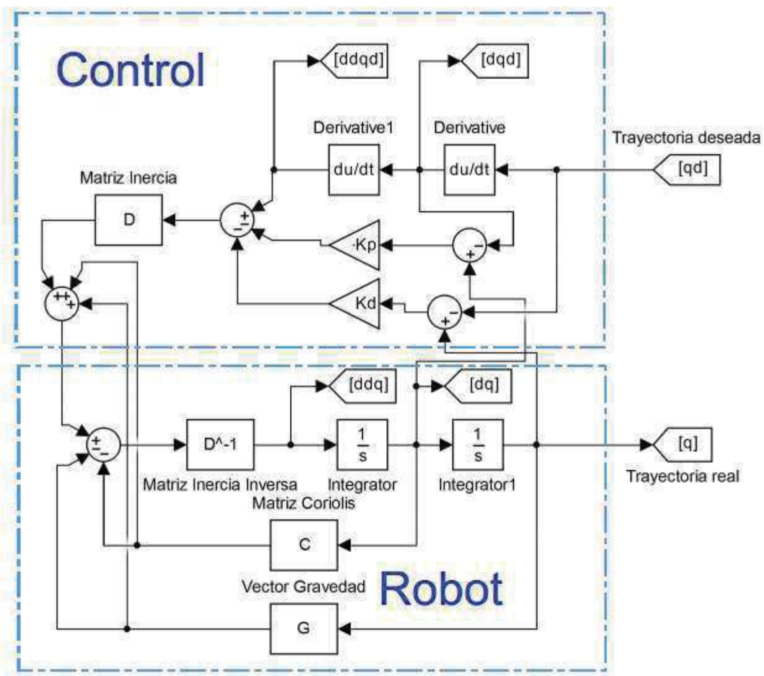


Fig. 3: Esquema gráfico del lazo de control.

II-C. Generación de trayectorias

La trayectorias requeridas q^{*} para el seguimiento del robot de dos grados de libertad, están dadas a través de las ecuaciones de Cooper [6]. Dicha relación matemática, es una propuesta aproximada para la descripción el movimiento relativo del sol respecto a la tierra con respecto al sistema coordenado horizontal. Los elementos para determinar dicho movimiento son

- Declinación solar (δ) que es el ángulo generado entre la dirección de los rayos solares y el ecuador.
- Día del año (n) el cual toma valores de 1 a 365.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 10, páginas 98 - 110.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

$$\delta = 23.45° \, sen\left(360°\left(\frac{248+n}{365}\right)\right) \quad (13)$$

La relación de Cooper provee, dado un día del año la condición inicial del movimiento solar.

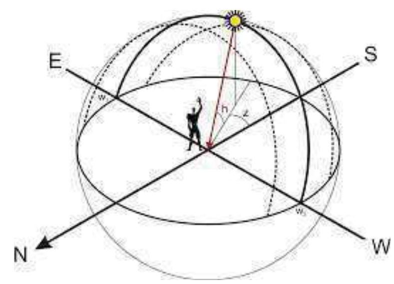


Fig. 4: Orientación solar.

Para la completa determinación de la posición solar, se requieren los ángulos respecto al ecuador en sistema coordenado esférico dadas la longitud L y latitud λ del sistema coordenado local (punto de observación).

- Ángulo horario τ, es el desplazamiento angular del Sol sobre el plano de la trayectoria solar dada una hora de observación.
  $\tau = (12 - hora)15°$, donde $hora = 0$ en el cenit.

- Ángulo de eje de elevación (h) es generado por el rayo solar en el plano vertical.

$$sen\,(h) = (\cos(\lambda)\cos(\delta)\cos(\tau)) + (sen(\lambda)sen(\delta))$$

- Ángulo de eje azimutal z es el que se forma de la proyección del sol sobre el plano horizontal con la dirección sur. Toma el valor de 0 cuando es el medio día.

$$cos\,(z) = \frac{sen(h)sen(\lambda) - sen(\delta)}{\cos(h)\cos(\lambda)}$$

Así los valores de h y z darán la trayectoria buscada q^* descrita como

$$q^* = \begin{bmatrix} q_1^* \\ q_2^* \end{bmatrix} = \begin{bmatrix} z \\ h \end{bmatrix} \quad (14)$$

III. Resultados numéricos

Utilizando el Software de diseño SolidWorks y el ToolBox- SimMechanics de Matlab se desarrolló un mecanismo equivalente al planteado, cuyas características se exportaron al entorno Simulink, donde se realizó la simulación numérica del controlador con la trayectoria propuesta. La figura 5 muestra el mecanismo realizado, en el entorno de SimMechanics, donde se aprecian los dos grados de libertad. Éste se encuentra compuesto por tres piezas, una base con masa de 1 kilogramo (kg), el eslabón 1 con una masa de 1 kg y finalmente, el eslabón 2 que agrega una pieza en su extremo equivalente a un PV, resultando así una masa total esta última pieza de 2 kg . La generación de trayectoria, así como el controlador propuesto se programaron en el entorno Matlab Simulink, haciendo el cálculo para el intervalo $t \in [0,10]$ s.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 10, páginas 98 - 110.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.



Fig. 5: Modelo equivalente en SimMechanics.

Las ganancias del controlador fueron las siguientes:

$$K_P = \begin{bmatrix} 500 & 0 \\ 0 & 40000 \end{bmatrix}, \qquad K_D \begin{bmatrix} 30 & 0 \\ 0 & 100 \end{bmatrix}$$

Con un periodo de muestreo de 0.0005 s. El esquema a bloques de la simulación se presenta en la figura 6.

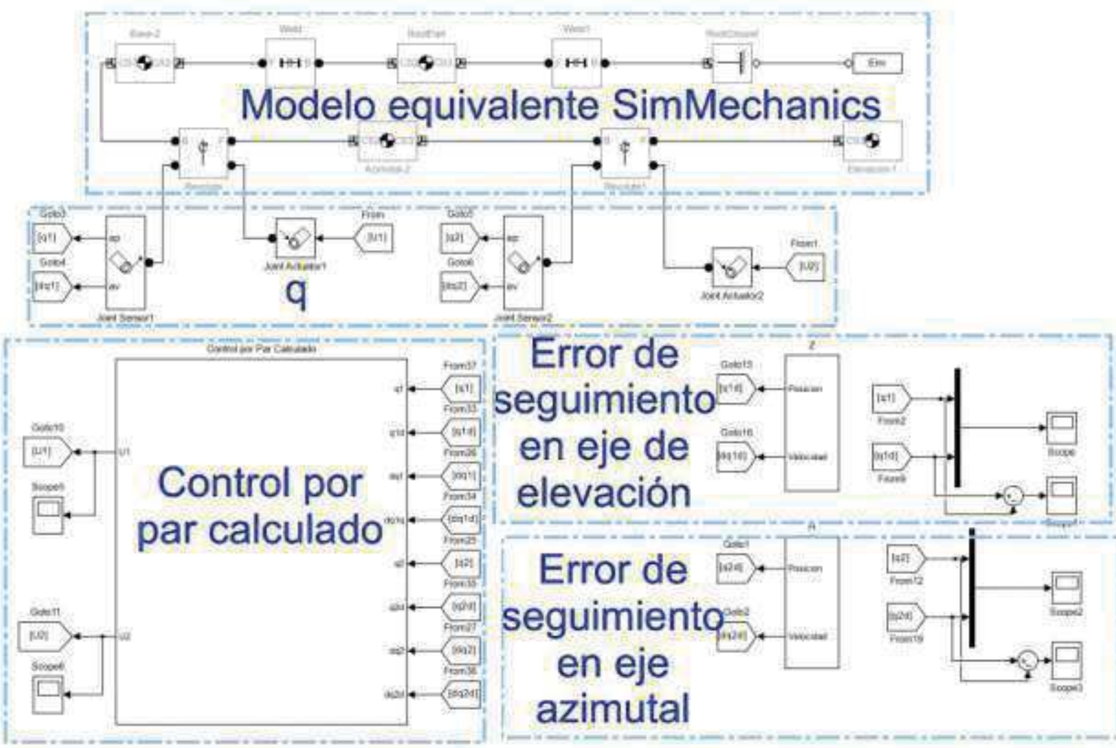


Fig. 6: Diagrama a bloques del programa de control en el entorno de SimMechanics.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

En las figuras 7 y 8 se muestran los resultados de seguimiento para los ejes azimutal y de elevación, empalmando las trayectorias reales y deseadas, las cuales se mantienen en una cercanía. Adicionalmente en las figuras 9 y 10 se muestra para las gráficas anteriores el error en cada eje respectivamente. Dado que el controlador se desarrolló íntegramente en Simulink, y el modelo realizado se exportó directamente del entorno de SolidWorks, el modelo obtenido está bien validado, lo cual permite una exportación a un entorno real con ajustes menores.



Fig. 7: Seguimiento de trayectoria eje azimutal.



Fig. 8: Seguimiento de trayectoria eje de elevación.



Fig. 9: Error de seguimiento (eje azimutal).

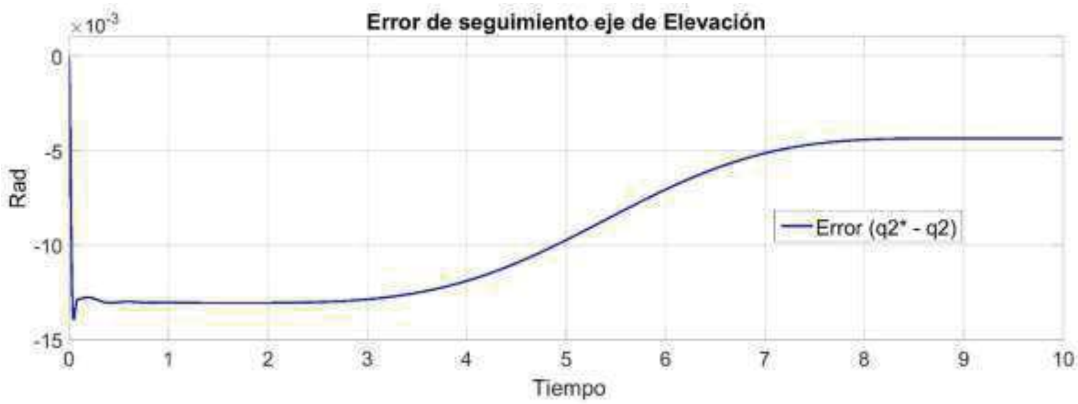


Fig. 10: Error de seguimiento (eje de elevación).

IV. Conclusiones

Se logró el modelado dinámico, generación de trayectoria solar, así como la implementación del control por par calculado de un sistema robótico seguidor solar de dos grados de libertad. La estrategia propuesta se comprobó mediante un esquema de simulación en SimMechanics, el cual permite estimar los pares cercanos a un esquema real al utilizar las propiedades mecánicas de los mecanismos involucrados. Como trabajo futuro se propone la implementación experimental de la estrategia propuesta, así como la implementación de estrategias de minimización energética en las tareas de control, así como trabajar en esquemas alternos de generación de trayectorias que incluyan entornos como nubes que puedan afectar las trayectorias deseadas reales.

Referencias

[1] C. Riba Romeva, *Recursos Energéticos y Crisis: el fin de 200*.Ediciones Octaedro, 2012.
[2] Grupo NAP, *Energía Solar Fotovoltaica*. Madrid, Colegio Oficial de Ingenieros de Telecomunicación, 2002.
[3] O. Perpiñan, *Energía Solar Fotovoltaica*. http://oscarperpinan.github.io/esf/, 2015.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

[4] A. Gonzalo y D. Valeria, *Manual Técnico de Energía Solar Volumen I: Fundamentos*. Publicaciones - Laboratorio de Energía Solar (LES), Universidad de la República de Uruguay, 2013.
[5] R. Kelly y V. Santibáñez, *Control de movimiento de robots manipuladores*. Pearson Education, SA, 2003.
[6] I. Reda y A. Andreas, "Solar position algorithm for solar radiation applications". Solar energy, 2004, vol. 76, no 5, pp. 577-589.
[7] M. H. M. Sidek, et al., "GPS based portable dual-axis solar tracking system using astronomical equation". IEEE International Conference on Power and Energy (PECon), 2014. IEEE, 2014. pp. 245-249.
[8] B. J. Huang, W. L. Ding y Y. C. Huang,"Long-term field test of solar PV power generation using one-axis 3-position sun tracker". Solar Energy, 2011, vol. 85, no. 9, pp. 1935-1944.
[9] H. Fathabadi, "Novel high efficient offline sensorless dual-axis solar tracker for using in photovoltaic systems and solar concentrators". Renewable Energy, 2016, vol. 95, pp. 485-494.
[10] R. S. Hartenberg, y J. Denavit, *Kinematic synthesis of linkages*. McGraw-Hill, 1964.
[11] M. W. Spong y M. Vidyasagar, *Robot dynamics and control*. John Wiley & Sons, 2008.
[12] R. P. Paul, *Robot Manipulators: Mathematics, Programing and Control*. The MIT Press, 1981.
[13] H. Sira-Ramírez, et al. *Control de sistemas no lineales: linealización aproximada, extendida, exacta*. Pearson Prentice Hall, 2005.

*Los autores agradecen a la Secretaria de Ciencia, Tecnología e Innovación (SECITI) por el apoyo recibido bajo el proyecto SECITI C-2016 272, así como a la Secretaría de Posgrado e Investigación del Instituto Politécnico Nacional a través del proyecto 20160354 y el programa de becas BEIFI.
[1,3] Instituto Politécnico Nacional-UPIITA. slash_sipr@hotmail.com, aluvianoj@ipn.mx
[2] Instituto Politécnico Nacional-CIDETEC. dfloreshe@ipn.mx
[4] Instituto Politécnico Nacional-ESIME Zacatenco. norma.lozada.c@gmail.com
[5] Instituto Politécnico Nacional-UPIBI. ichairezo@gmail.com

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 11, páginas 111 - 120.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Synthesis and analysis of a fractional-order sliding mode control for mechanical systems

Carlos A. Rodriguez[1], Luis T. Aguilar[1], Alejandra Ferreira[2], and Eusebio Bugarín[3]

[1] Instituto Politécnico Nacional—CITEDI, Avenida Instituto Politécnico Nacional No. 1310 Colonia Nueva Tijuana, Tijuana 22435 México (emails: crodriguez@citedi.mx, laguilarb@ipn.mx).
[2] CONACyT Research Fellow at Instituto Politécnico Nacional—CITEDI, Avenida Instituto Politécnico Nacional No. 1310 Colonia Nueva Tijuana, Tijuana 22435 México (email: dferreira@citedi.mx).
[3] Tecnológico Nacional de México-Instituto Tecnológico de Ensenada, Blvd. Tecnológico No. 150, Col. Ex Ejido Chapultepec, Ensenada, B. C., México, C. P. 22780 (email: ebugarin@ite.edu.mx).

Abstract— **Synthesis and analysis of a fractional-order sliding mode control (FOSMC) for a perturbed double integrator are addressed. The novelty of the paper resides in that a fractional-order sliding mode surface, with order less than one, is proposed to ensure the asymptotic stability of a second order system. Compared with conventional sliding mode scheme, the proposed method decreases the frequency and the amplitude of the control signal (also known as chattering); consequently, the overheating and deterioration in mechanical components of the actuator may be reduced. Despite dynamical models described by fractional-order integro-differential equations allow a straightforward synthesis of fractional-order controllers, in this paper the integer-order differential equation of the plant is used instead of its fractional version thus avoiding an extra step in the control design. Moreover, it is shown that the use of a fractional sliding mode controller attenuates the chattering, preserving the robustness and finite-time reachability properties of sliding mode controllers. Numerical and experimental results demonstrate the effectiveness of the proposed method in an uncertain DC motor with friction.**

Introduction

Fractional-order systems and controllers are an emerging research field in engineering problems. Fractional calculus, with more than 300 years old of history, is a generalization of ordinary differentiation and integration of arbitrary order. During past three centuries, this subject was devoted to mathematicians. In the last few years, however fractional calculus has been-applied to several fields of engineering, science and economics [1].

In the last decades the use of fractional-order integro-differential equations has attracted considerable research interest for nonlinear control design and analysis for complex systems including infinite-dimensional dynamical systems. Some physical systems which are better modelled by fractional-order differential equations are irrigation systems [2], diffusion processes [3], ionic polymer-metal composite actuators [4], among others (see, e.g., [5]). The extra degrees-of-freedom from using of fractional-order integrator and differentiator made possible to further improve the performance in comparison with traditional integer order controller.

Robust control design for plants governed by fractional-order differential equations has become a developing area for the control community. For example, Jakovljevic *et al*. [6] deals with applications of sliding-mode-based fractional control techniques to address tracking and stabilization control tasks for some classes of nonlinear uncertain fractional-order systems. Kamal and Bandyopadhyay [7] design a globally exponentially stable controller for fractional-order chain of integrators based on contraction theory. Aguila-Camacho and Duarte-Mermoud [8] make an analysis of three classes of fractional differential equations appearing in the field of fractional adaptive system where Caputo definition [9] plays an important role in the analysis. Tepljakov *et al*. [10] make applications for a mechanical system using a fractional PID controller.

Concerning variable structure controllers, which are recognized by its robustness properties, Vinagre and Calderon [11] propose a controller with the fractional-order control law and sliding surface for a double integrator structure. Huang et al. [12] apply the fractional calculus into the sliding surface design for the speed control of a permanent magnet synchronous motor. Muñoz-Vazquez *et al*. [13] introduce a fractional sliding surface of commensurable rational order r to control Euler-Lagrange systems. Efe [14] propose the fractional order sliding mode control with the reaching law approach. Summarizing, the FOSMC achieves better control performance with smaller chattering level with respect to conventional sliding mode controllers but preserving the robustness against external load disturbance and parameter variations [15], [16].

In [17] Zhang establishes the three main approaches to counteract the chattering phenomenon in sliding mode control (SMC) systems proposed in the last two decades: continuous approximation of the relay or introduce bounder layer near the sliding surface [18], use an asymptotic state-observer to confine chattering in the observer dynamics bypassing the plan [19], use higher-order or second-order sliding mode

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

control (SOSMC) algorithms [20]. However, the drawbacks of the continuous approximations and of the observed-based approach are the deterioration of accuracy and systems robustness, respectively.

Sliding mode controllers are well known for its ability to deal with uncertain systems. Nevertheless, SMC are tarnished by the so-called chattering effect. Chattering may be destructive to real systems, thus, the use of a continuous approximation of signum function is considered to counteract its effects. This approach attenuates chattering at the expenses of a lost of robustness. Fractional order controllers, however, reduce the chattering while preserving robustness properties.

In this work, the synthesis and analysis of a fractional-order sliding mode control for a perturbed double integrator was addressed. In contrast to the above mentioned relevant literature, the novelty of the paper resides in that a fractional-order sliding mode surface, with positive order less than one, is proposed to ensure asymptotic stability of the equilibrium point of the double integrator. Moreover, the idea behind the paper is to increase the frequency and decrease the amplitude of chattering in the control input, , without losing the robustness and finite-time reachability properties of the sliding mode controllers. Despite fractional-order controllers can be straightforwardly synthesized from dynamical models described by fractional-order integro-differential equations. In this paper the integer-order differential equation of the model of the plant was used instead of its fractional version thus simplifying the control design.

This paper is organized as follows. We start in Section II, recalling basic definitions about fractional-order derivatives and integrals. In Section III, the problem statement and synthesis and analysis of the fractional-order sliding mode control is developed for a perturbed double integrator. In Section IV, the performance of the controller is evaluated in a numerical and experimental study with a CD motor. Finally, conclusions are provided in Section V.

Notation: We let $\mathbb{N}$ denote the set of positive natural numbers and $\mathbb{N}_0$ denote the set of non-negative integers. We let $\mathbb{R}$ and $\mathbb{C}$ denote the set of real and complex numbers, respectively.

I. Basic Definitions

In this section we present definitions which play an important role in theoretical part of the paper. They will be used in the analysis of the forthcoming section dealing with the synthesis of the fractional-order sliding mode controller.

The fractional-order derivative operator, denoted by D , of order is defined as

$$\mathrm{D}^{\lambda} \;\square\; {}_a\mathrm{D}_t^{\lambda} = \begin{cases} \dfrac{d^{\lambda}}{dt^{\lambda}} & \mathrm{R}(\lambda) > 1 \\ 1 & \mathrm{R}(\lambda) = 0 \\ \int_a^t (dt)^{-\lambda} & \mathrm{R}(\lambda) < 1 \end{cases} \tag{1}$$

where a and t are the limits of the operator. Generally, $\alpha \in \square$.

A. Definitions of Fractional-Order Derivatives and Integrals

Throughout, the following definition for the fractional-order derivatives, introduced by Caputo [8], will be used:

$${}_a^C\mathrm{D}_t^{\alpha} f(t) = \mathrm{D}^{\alpha-n}\mathrm{D}^{n} f(t) = \frac{1}{\Gamma(\alpha-n)}\int_a^t \frac{f^{(n)}(x)}{(t-x)^{\alpha-n+1}}dx \tag{2}$$

where $f(x)$ with $x \in \mathbb{R}$ is an arbitrary integrable function, $\Gamma(\alpha-n)$ is the gamma function, $0 \le n-1 \le \alpha < n$, and $t > 0$. According to the latter definition, the fractional-order derivative of a real number C is

$$\mathrm{D}^{\alpha} C \equiv 0, \qquad \alpha > 0, \tag{3}$$

and the following property

$$\mathrm{D}^{\alpha} C f(x) = C\mathrm{D}^{\alpha} f(x), \qquad \alpha > 0, \tag{4}$$

also holds.

The Caputo's definitions is the most used for fractional-order integrals and derivatives for its relevance in automatic control because it takes into account the initial conditions of the system [21] been realistic from the physical point of view. Another advantage of the above definition is that initial conditions of its Laplace transformation have the same shape than the integer-order differential equations. Moreover,

derivatives and integrals of fractional-order provide a powerful tool for the description of properties of heritage and memory of different substances [22], [23], being an advantage with respect to the integer-order analysis.

According to Caputo's definition, the fractional-order derivative can also be expressed as the combination of an integer-order derivative and a fractional-order integral, that is,

$$\mathrm{D}^{\alpha} f(t) \triangleq \mathrm{I}^{\,m-\alpha} \mathrm{D}^{m} f(t), \tag{5}$$

where I denotes the fractional-order integral operator and m is a positive integer value. The Riemann-Liouville definition [24] is considered for the fractional-order integral that is

$$\mathrm{I}^{\,\alpha} f(t) = \frac{1}{\Gamma(\alpha)} \int_a^t (t-x)^{\alpha-1} f(x) dx. \tag{6}$$

Thus, the Laplace transformation of Caputo fractional-order derivative of a function $f(t): \mathbb{R} \mapsto \mathbb{R}$ is given by

$$\begin{aligned} \mathrm{L}\left\{\mathrm{D}^{\alpha} f(t)\right\} &= \mathrm{L}\left\{\mathrm{I}^{\,m-\alpha} \mathrm{D}^{m} f(t)\right\} \\ &= s^{-(m-a)} \left[s^m F(s) - \sum_{k=0}^{m-1} s^{m-k-1} f^{(k)}(0) \right] \\ &= s^{\alpha} F(s) - \sum_{k=0}^{m-1} s^{\alpha-k-1} f^{k}(0), \end{aligned} \tag{7}$$

with $m-1 \le \alpha < m.$

B. Mittag-Leffler Function

The Mittag-Leffler function defined as

$$E_{\beta,\eta}^{(m)}(z) = \sum_{k=0}^{\infty} \frac{(k+m)!}{k!} \frac{z^k}{\Gamma(\beta k + \beta m + \eta)}, \tag{8}$$

is a generalization of the Euler function for fractional-order integro-differential equations. Here, m is the integer order degree of Mittag-Leffler function derivation, $z \in \mathbb{C}$, and parameters β and η are the fractional-orders of the function derivation.

The procedure for finding a solution of a fractional-order integro-differential equation, consists first into apply the Laplace transform to the equation in question and then group it into terms according to the number of poles of the system. Finally, the solution, which depends on the Mittag-Leffler function, is obtained after using Laplace anti-transformation.

II. FOSMC OF A PERTURBED DOUBLE INTEGRATOR

The purpose of this Section is to synthesize a fractional-order sliding mode controller for a perturbed rotational double integrator given by

$$\ddot{q} = u + w \tag{9}$$

where $q(t) \in \mathbb{R}$ denotes the angular position, $\dot{q}(t) \in \mathbb{R}$ is the angular velocity, $u(t) \in \mathbb{R}$ is the control input, $t \ge 0$ is the time variable, and $w(t) \in \mathbb{R}$ is the external disturbance.

The *control objective* is to drive the position $q(t)$ and the velocity $\dot{q}(t)$ to the origin, that is

$$\lim_{t\to\infty} \left\| \begin{bmatrix} q(t) \\ \dot{q}(t) \end{bmatrix} \right\| = 0 \tag{10}$$

for arbitrary initial conditions $q(0)$, $\dot{q}(0)$ in spite of the presence of external perturbations.

The following fractional-order sliding mode controller

$$u(t) = -n\mathrm{D}^{\frac{2-\alpha}{2}} \dot{q}(t) - m\mathrm{D}^{1-\alpha} \dot{q}(t) - \rho \mathrm{I}^{\,\delta} \operatorname{sign}(S(t)), \tag{11}$$

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

is proposed to satisfy the control objective where S is the fractional-order sliding surface defined as

$$S = \mathrm{D}^{\alpha} q(t) + n\mathrm{D}^{\beta} q(t) + mq(t), \tag{12}$$

with $m > 0$, $0 < \alpha < 1$, $0 < \beta < 1/2$, and $0 < \delta < 1$.

A. Stability Analysis of the Origin

First, we need to verify that trajectories tends to the origin once the trajectories reach the discontinuity set $S = \{(q,\dot{q})^T \in \sqcup^2 : S(q) = S(\dot{q}) = 0\}$ that is $(q,\dot{q})^T \in S$. In other words, we need to show that the solution of the fractional-order differential equation

$$\mathrm{D}^{\alpha} q(t) + n\mathrm{D}^{\beta} q(t) + mq(t) = 0 \tag{13}$$

under the initial conditions $q(0) = b_1$ and $\dot{q}(0) = b_2$, reach the origin asymptotically. It is assumed that trajectories are initialized within the set S, that is, $(q(0),\dot{q}(0))^T \in S$.

By applying the Laplace transformation to (14) we have

$$s^{\alpha}Q(s) - s^{\alpha-1}q(0) - s^{\alpha-2}\dot{q}(0) + ns^{\beta}Q(s) - ns^{\beta-1}q(0) + mQ(s) = 0$$

then,

$$Q(s) = \frac{b_1 s^{\alpha-1} + b_2 s^{\alpha-2} + b_1 n s^{\beta-1}}{s^{\alpha} + ns^{\beta} + m}. \tag{14}$$

For simplicity, let us define α = 2β therefore the latter transfer function can be expressed as

$$Q(s) = \frac{b_1 s^{2\beta-1} + b_2 s^{2\beta-2} + b_1 n s^{\beta-1}}{(s^{\beta} + \lambda_1)(s^{\beta} + \lambda_2)}. \tag{15}$$

where $\lambda_i = n/2 \pm \sqrt{n^2 - 4m}/2 \ (i = 1,2)$. In consequence

$$\begin{aligned} s_1 &= \left(-\frac{1}{2}n - \frac{1}{2}\sqrt{n^2 - 4m}\right)^{1/\beta} \\ s_2 &= \left(-\frac{1}{2}n + \frac{1}{2}\sqrt{n^2 - 4m}\right)^{1/\beta}. \end{aligned} \tag{16}$$

Hence, the poles will be negative for any $n > 0$ and any $\beta > 0$ satisfying the condition $\beta = (2k+1)(2p+1)$ with $p > k$ $(p \in \mathbb{N}, k \in \mathbb{N}_0)$ been consistent with the condition given in (13).

Remark 1: The solution of the fractional-order differential equation (14) given by

$$\begin{aligned} q(t) = &-\frac{\lambda_1}{\lambda_2 - \lambda_1}\left[b_1 E_{\beta,1}(-\lambda_1 t^{\beta}) + b_2 t E_{\beta,2}(-\lambda_1 t^{\beta})\right. \\ &\left.+ b_1 n t^{\beta} E_{\beta,\beta+1}(-\lambda_1 t^{\beta})\right] + \frac{\lambda_2}{\lambda_2 - \lambda_1}\left[b_1 E_{\beta,1}(-\lambda_2 t^{\beta})\right. \\ &\left.+ b_2 t E_{\beta,2}(-\lambda_2 t^{\beta}) + b_1 n t^{\beta} E_{\beta,\beta+1}(-\lambda_2 t^{\beta})\right] \end{aligned}$$

is obtained from the anti-transformation of $Q(s)$, that is

$$\begin{aligned} \mathrm{L}^{-1}\{Q(s)\} = &\ \mathrm{L}^{-1}\{-\frac{\lambda_1}{\lambda_2 - \lambda_1}\frac{b_1 s^{\beta-1} + b_2 s^{\beta-2} + b_1 n s^{-1}}{(s^{\beta} + \lambda_1)}\} \\ &+ \mathrm{L}^{-1}\{\frac{\lambda_2}{\lambda_2 - \lambda_1}\frac{b_1 s^{\beta-1} + b_2 s^{\beta-2} + b_1 n s^{-1}}{(s^{\beta} + \lambda_2)}\} \end{aligned}$$

where the continuous Laplace anti-transformation of Mittag-Leffler function by

$$\mathrm{L}^{-1}\left\{\frac{m!s^{\beta-\eta}}{(s^{\beta}+a)^{m+1}}\right\}=t^{\beta m+\eta-1}E^{(m)}_{\beta,\eta}(-at^{\beta}) \tag{17}$$

was used. Since $\lim_{t\to\infty}E_{\beta,\eta}(-at^{\beta})=0$ then $q(t)$ tends to the origin as t→∞.

B. Existence of Sliding Modes

The conditions for the trajectories to converge to the manifold $S(q)$ = 0 and the sliding manifold to exists on this manifold may be derived based on the Lyapunov function [25]

$$V=\frac{1}{2}S^{2} \tag{18}$$

whose time derivative along the solution of the closed-loop system (9), (12) is given by

$$\begin{aligned}\dot V = S\dot S &= S\left(\mathrm{D}^{\alpha}\dot q(t)+n\mathrm{D}^{\beta}\dot q(t)+m\dot q(t)\right)\\ &= S\left(\mathrm{D}^{\alpha-1}\ddot q(t)+n\mathrm{D}^{\beta}\dot q(t)+m\dot q(t)\right).\end{aligned}$$

Substituting the right-hand side of (9) and the control input (12) into the last equation we get

$$\begin{aligned}\dot V &= S\left[\mathrm{D}^{\alpha-1}\left(-n\mathrm{D}^{\frac{2-\alpha}{2}}\dot q-mD^{1-\alpha}\dot q-\rho\mathrm{I}^{\delta}\,\mathrm{sign}(S)+w\right)+n\mathrm{D}^{\beta}\dot q+m\dot q\right]\\ &= S\left[-n\mathrm{D}^{\frac{\alpha}{2}}\dot q-m\dot q-\rho\mathrm{D}^{\alpha-1}\mathrm{I}^{\delta}\,\mathrm{sign}(S)+\mathrm{D}^{\alpha-1}w+n\mathrm{D}^{\beta}\dot q+m\dot q\right]\\ &= S\left[-\rho\mathrm{D}^{\alpha-1}\mathrm{I}^{\delta}\,\mathrm{sign}(S)+\mathrm{D}^{\alpha-1}w\right]\end{aligned}$$

Since $\alpha-1-\delta$ and $\alpha-1$ are negatives, the following holds (cf. [21, p. 88])

$$\begin{aligned}\dot V &= \mathrm{D}^{\alpha-1-\delta}\left[-\rho\,\mathrm{sign}(S)+\mathrm{D}^{\delta}w\right]S\\ &\le -\mathrm{D}^{\alpha-1-\delta}\left[\rho-\left|\mathrm{D}^{\delta}w\right|\right]|S|.\end{aligned} \tag{19}$$

Comment: $\mathrm{D}^{\delta}w(t)$ is the fractional derivative of external disturbance which is uniformly bounded by a constant W_1.

Therefore, $\dot V$ proofs to be negative definite for any ρ satisfying

$$\left|\mathrm{D}^{\delta}w\right|<W_1<\rho \tag{20}$$

where W_1 is a strict positive constant fixed a priori.

C. Finite-Time Reachability

Let us take the last inequality (20)

$$\frac{dV}{dt}\le -\mathrm{D}^{\alpha-1-\delta}\left[\rho-W_1\right]|S|. \tag{21}$$

From (19) we have $|S|=\sqrt{2V}$, then the last inequality can be rewritten as

$$\frac{dV}{dt}\le -\sqrt{2}\mathrm{D}^{\alpha-1-\delta}\left[\rho-W_1\right]\sqrt{V} \tag{22}$$

By rewriting last equation as

$$\frac{dV}{\sqrt{V}}\le -\sqrt{2}\mathrm{D}^{\alpha-1-\delta}\left[\rho-W_1\right]dt \tag{23}$$

and integrating from 0 to t_f it follows that

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

$$-2\sqrt{V(0)} \le -\sqrt{2}(\rho - W_1) \cdot \mathrm{D}^{\alpha-1-\delta}(1). \tag{24}$$

By using definition (6) we have

$$-2\sqrt{V(0)} \le -\frac{\sqrt{2}(\rho - W_1)}{2+\delta-\alpha} \cdot \frac{t_f^{2+\delta-\alpha}}{\Gamma(2+\delta-\alpha)}. \tag{25}$$

Finally, the estimated time that trajectories reach the sliding surface is given by

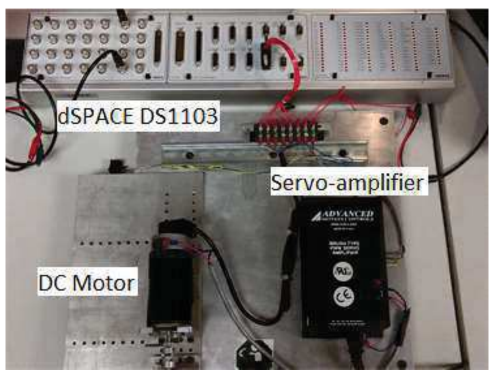
$$t_f \le \left(\frac{2(2+\delta-\alpha)\Gamma(2+\delta-\alpha)\sqrt{V(0)}}{\sqrt{2}(\rho - W_1)} \right)^{1/(2+\delta-\alpha)}. \tag{26}$$

III. Experimental Study

Performance issues of the fractional order sliding mode controller were tested numerically and experimentally on a DC motor with friction (see Fig. 1). For the experiments we used the test-bed installed in the robotics and control laboratory of CITEDI-IPN which involves a DC motor manufactured by LEADSHINE, capable to handle high-switching frequency, and the dSPACE control board DS1103.

The controller was implemented using SIMULINK 2007 from MathWorks running on a personal computer with AMD A4-3400, 2.70 GHz, 2 GB processor. The resolution of the encoder is 1000 counts/rev. The amplifier of the motor accepts a control input from the D/A converter in the range of $\pm$ 10 V. The sampling time was 10^{-3} s.

Figure 1. Experimental testbench.



The DC motor is governed by the following differential equation

$$J\ddot{q} + f_v\dot{q} + f_c\,\mathrm{sign}(\dot{q}) = \tau \tag{27}$$

where $\dot{q}(t) \in \mathbb{R}$ is the angular velocity of the motor shaft, $\tau(t)$ is the control input, $J = 3.21\times10^{-5}$ Kgm^2 is the inertia of the motor shaft, $f_v = 1.35\times10^{-5}$ Nms·rad^{-1} is the coefficient of viscous friction, and $f_c = 6.8\times10^{-3}$ Nm is the coefficient of Coulomb friction. Therefore, the implemented control input was

$$\tau = Ju \tag{28}$$

where $u(t) \in \mathbb{R}$ is given in (12). The gain and parameter of the fractional order sliding mode controller were set to $m = 2$, $n = 8$, $\rho = 0.1$, $\alpha = 0.78$, $\beta = 0.39$, and $\delta = 0.2$. These gains were chosen ad-hoc for avoiding the resonance frequency in the motor.

A. *Experimental Results*

To implement fractional order sliding mode control the fractional-order operators are approximated by an integer order transfer functions. Let $[\omega_A, \omega_B]$ be the frequency range of interest, then the approximation of the frequency-band fractional-order operators can be determined as follows [27]:

$$s^{\lambda} \approx \left(\frac{\omega_u}{\omega_h}\right)^{\lambda} \prod_{k=-N}^{N} \frac{1+\dfrac{s}{\omega_k'}}{1+\dfrac{s}{\omega_k}} \tag{29}$$

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

with

$$\omega_k' = \omega_b \left(\frac{\omega_h}{\omega_b} \right)^{\frac{k+N+\frac{1}{2}(1-\lambda)}{2N+1}}, \ \omega_k = \omega_b \left(\frac{\omega_h}{\omega_b} \right)^{\frac{k+N+\frac{1}{2}(1+\lambda)}{2N+1}}$$

where ω_k' is the zero of rank k, ω_k is the pole of rank k and $\omega_u = \sqrt{\omega_b \omega_h}$. Here, $\omega_b \ll \omega_A$ and $\omega_h \gg \omega_B$ are the low-transitional and high-transitional frequencies, respectively.

Figure 2. Numerical joint position and control input for the disturbance-free closed-loop system

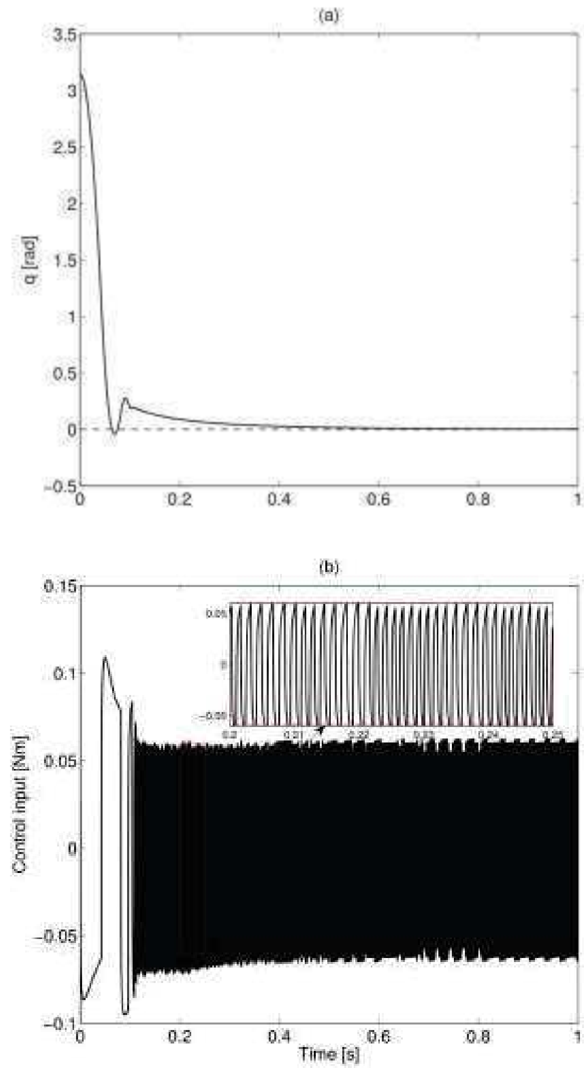
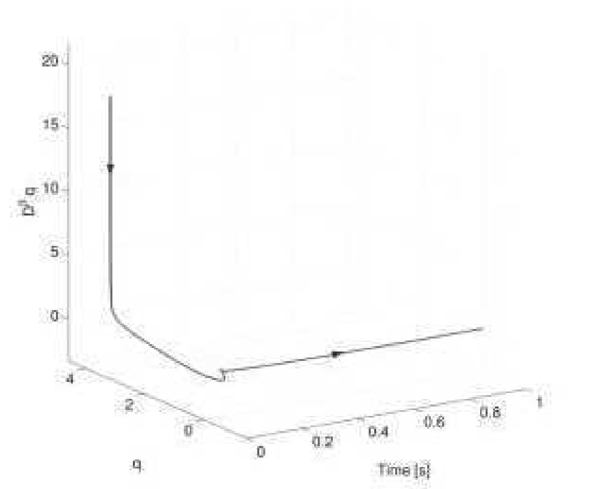


Figure 3. Phase portrait of the closed-loop system (28), (29).

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

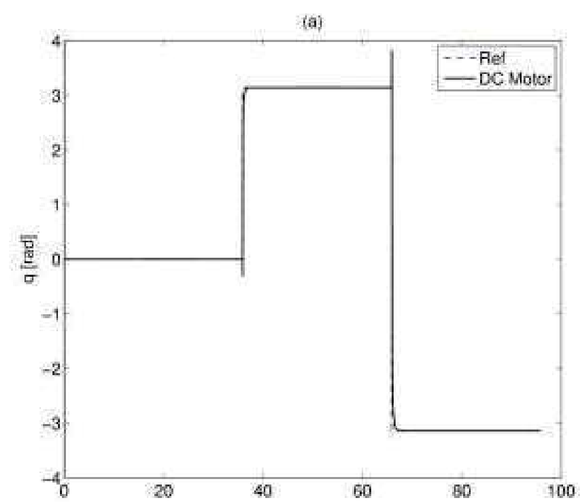


For the switching frequency of the DC motor, given by the manufacturer, the required transitional frequencies for the approximation of the fractional-order operators are set to $\omega_b = 2$ rad s^{-1} and $\omega_h = 2\times10^4$ rad·s^{-1}. Figure 2(a) shows the response of the DC motor, under arbitrary initial condition $q(0) = \pi$ rad and $\dot{q}(0) = 0$ rad·s^{-1}, where it is corroborated that the joint position reaches the origin asymptotically. Figure 2(b) also illustrates the behavior of the control input where the switching frequency is less than the resonance frequency thus minimizing its undesired effect in the actuator like the overheating of components. Figure 3 depicts the phase-portrait of the closed-loop system. This figure shows that the trajectory reaches the sliding surface at finite-time $t_f = 1.33$ seconds satisfying the estimated reachability time given by inequality (27), that is, $t_f \leq 67.44$ seconds.

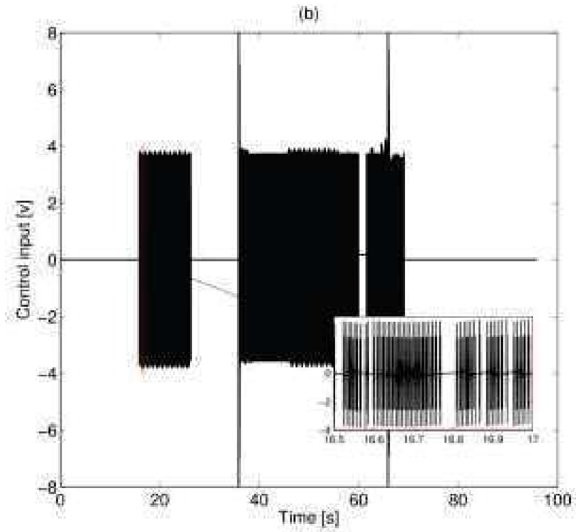
Fig. 4(a) shows the response of the closed-loop system affected by an external bounded perturbation $w(t)=0.05\sin(2\pi ft)$. Notice that the joint position follows the reference trajectory in spite of the presence of external disturbance. Figure 4(b) also shows the control input.

Figure 5 illustrates comparative control inputs responses of the closed-loop system using fractional-order sliding surface (13) versus a conventional sliding mode control, that is, $u = -m\dot{q} + \rho\,\text{sign}(S)$ with $S = \dot{q} + 50q$ and $\rho > W^+$, where $w(t)$ is an external disturbance which is uniformly bounded by a constant $W^+ > 0$, fixed a priori.

Figure 4. Experimental FOSMC responses of the DC motor under external disturbance.



ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

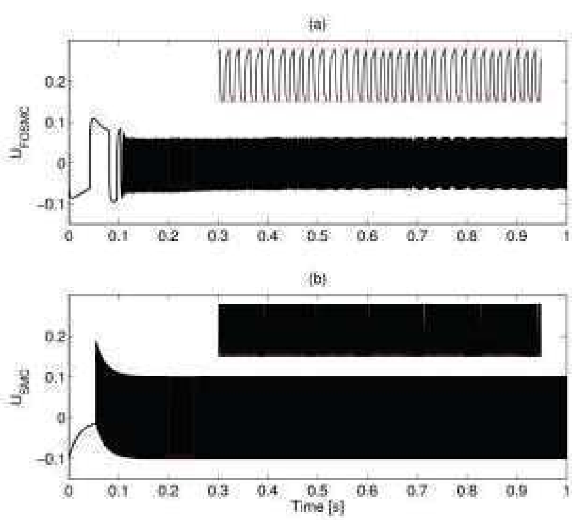


Notice that frequency of chattering of first order sliding mode control is higher than the chattering frequency of fractional-order sliding mode control. Therefore with a fractional-order sliding surface the effects caused by chattering can be minimized also preserving the finite-time reachability and robustness properties against external perturbations with bounded fractional derivative.

IV. Future Works

We addressed the synthesis and analysis of a fractional-order sliding mode control for a perturbed double integrator. Robustness against external disturbances with a priori known norm bounds and finite-time reachability were analytically verified thus demonstrating that well-known properties of sliding mode controllers are preserved. Numerical and experimental results corroborate the effectiveness of the proposed control strategy. The resulting controller reduces the frequency of chattering, ensuring to be less than the resonant frequency. In addition, matched disturbances rejection and finite-time reachability to the sliding manifold is demonstrated in spite of the presence of Coulomb and viscous friction. Some tools for numerical implementation of fractional-order differential equations were also provided. Analytical verification of chattering attenuation via frequency domain methods still in progress.

Figure 5. Comparative responses of the control inputs of the closed-loop system using (a) fractional-order sliding surface versus (b) integer-order sliding surface $(S = \dot{q} + cq,\ c > 0)$.



Acknowledgments

L. Aguilar gratefully acknowledge the financial support from Consejo Nacional de Ciencia y

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Tecnología (CONACYT) under Grant 268364.

REFERENCES

[1] D. Liwei, S. Shenmin, and G. Yong, "Attitude Control of Five Degrees of Freedom Air-Bearing Platform Based on Fractional Order Sliding Mode," in *2013 Third International Conference on Instrumentation, Measurement, Computer, Communication and Control*, 2013, pp. 1530–1534.
[2] S. N. Calderon-Valdez, V. Feliu-Batlle, and R. Rivas-Perez, "Fractional-order mathematical model of an irrigation main canal pool," *Spanish J. Agric. Res.*, vol. 13, no. 3, p. e0212, Aug. 2015.
[3] D. Sierociuk, T. Skovranek, M. Macias, I. Podlubny, I. Petras, A. Dzielinski, and P. Ziubinski, "Diffusion process modeling by using fractional-order models," *Appl. Math. Comput.*, vol. 257, pp. 2–11, 2015.
[4] R. Caponetto, S. Graziani, V. Tomasello, and A. Pisano, "Identification and Fractional Super-Twisting Robust Control of IPMC Actuators," *Fract. Calc. Appl. Anal.*, vol. 18, no. 6, pp. 1358–1378, Jan. 2015.
[5] C. A. Monje, Y. Chen, B. M. Vinagre, D. Xue, and V. Feliu, *Fractional-order Systems and Controls*. London: Springer London, 2010.
[6] B. Jakovljević, A. Pisano, M. R. Rapaić, and E. Usai, "On the sliding-mode control of fractional-order nonlinear uncertain dynamics," *Int. J. Robust Nonlinear Control*, vol. 26, no. 4, pp. 782–798, Mar. 2016.
[7] S. Kamal, B. Bandyopadhyay, and S. Spurgeon, "Stabilization of a fractional-order chain of integrators: a contraction-based approach," *IMA J. Math. Control Inf.*, vol. 32, no. 2, pp. 291–303, Jun. 2015.
[8] N. Aguila-Camacho and M. A. Duarte-Mermoud, "Boundedness of the solutions for certain classes of fractional differential equations with application to adaptive systems," *ISA Trans.*, vol. 60, pp. 82–88, 2016.
[9] M. Caputo, "Linear Models of Dissipation whose Q is almost Frequency Independent--II," *Geophys. J. Int.*, vol. 13, no. 5, pp. 529–539, Nov. 1967.
[10] A. Tepljakov, E. A. Gonzalez, E. Petlenkov, J. Belikov, C. A. Monje, and I. Petráš, "Incorporation of fractional-order dynamics into an existing PI/PID DC motor control loop," *ISA Trans.*, vol. 60, pp. 262–273, 2016.
[11] B. M. Vinagre and A. J. Calderon, "On Fractional Sliding Mode Control," in *7th Portuguese Conference on Automatic Control*, 2006.
[12] J. Huang, H. Li, F. Teng, and D. Liu, "Fractional order sliding mode controller for the speed control of a permanent magnet synchronous motor," *2012 24th Chinese Control Decis. Conf.*, pp. 1203–1208, 2012.
[13] A. J. Munoz Vazquez, V. Parra Vega, and A. Sanchez, "Free-model fractional-order absolutely continuous sliding mode control for euler-lagrange systems," in *53rd IEEE Conference on Decision and Control*, 2014, pp. 6933–6938.
[14] M. O. Efe, "Fractional order sliding mode control with reaching law approach," *Turkish J. Electr. Eng. Comput. Sci.*, vol. 18, no. 5, pp. 731–747, 2010.
[15] A. Mujumdar, S. Kurode, and B. Tamhane, "Fractional order sliding mode control for single link flexible manipulator," in *2013 IEEE International Conference on Control Applications (CCA)*, 2013, pp. 288–293.
[16] M. L. Corradini, R. Giambò, and S. Pettinari, "On the adoption of a fractional-order sliding surface for the robust control of integer-order LTI plants," *Automatica*, vol. 51, pp. 364–371, Jan. 2015.
[17] B. Zhang, Y. Pi, and Y. Luo, "Fractional order sliding-mode control based on parameters auto-tuning for velocity control of permanent magnet synchronous motor.," *ISA Trans.*, vol. 51, no. 5, pp. 649–56, Sep. 2012.
[18] I.-C. In-Cheol Baik, K.-H. Kyeong-Hwa Kim, and M.-J. Myung-Joong Youn, "Robust nonlinear speed control of PM synchronous motor using boundary layer integral sliding mode control technique," *IEEE Trans. Control Syst. Technol.*, vol. 8, no. 1, pp. 47–54, 2000.
[19] A. Bondarev, S. Bondarev, N. Kostylyeva, and N. Utkin, "Sliding modes in systems with asymptotic state observers," *Autom. i Telemech. (Automation Remote Control.*, vol. 46, no. 5, pp. 679–684, 1985.
[20] G. Bartolini, A. Pisano, and E. Usai, "On the second-order sliding mode control of nonlinear systems with uncertain control direction," 2009.
[21] F. J. Castillo, "Metodología de Diseño de Reguladores de Orden Fraccional a partir de Especificaciones Temporales. Aplicación a Canales Principales de Riego.," Castilla-La Mancha, 2010.
[22] I. Podlubny, "Fractional-Order Systems and Fractional-Order Controllers," Kosice, 1994.
[23] B. M. Vinagre, I. Podlubny, A. Dorcak, L., and V. Feliu, "On Fractional PID Controllers: A Frequency Domain Approach," in *Proceedings of IFAC Workshop on Digital Control - PID'00*, 2000.
[24] J. Liouville, "Mémorie sur l'intégration de l'équation (mx2 + nx +p)d^2y=dx^2 + (qx + r)dy=dx + sy = 0 á l'aide des différentielles á indice quelconque," *J. l'école Polytech.*, vol. 13, no. 21, pp. 162–186, 1832.
[25] V. I. Utkin, *Sliding Modes in Control and Optimization*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1992.
[26] I. Podlubny, *Fractional differential equations*, 1st ed. San Diego: Academic Press, 1999.
[27] A. Oustaloup, F. Levron, B. Mathieu, and F. M. Nanot, "Frequency-band complex noninteger differentiator: characterization and synthesis," *IEEE Trans. Circuits Syst. I Fundam. Theory Appl.*, vol. 47, no. 1, pp. 25–39, 2000.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 12, páginas 121 - 132.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Micro algoritmos genéticos en arquitecturas de sistemas embebidos para la autonomía de marcha en robots

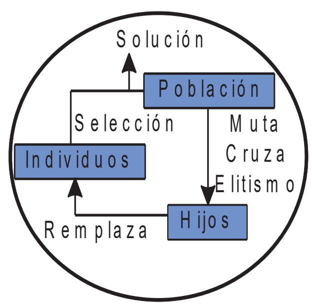
F. A. Chávez Estrada, J. C. Herrera Lozada, J. Sandoval Gutiérrez y M. I. Cervantes Valencia

***Resumen*— Los algoritmos genéticos (AG) aplicados en los sistemas de control han sido utilizados para incrementar la confiabilidad y desempeño de los procesos en la ingeniería; sin embargo el alto costo computacional y tiempo de ejecución requerido para dotar de autonomía la marcha de un sistema robótico con recursos limitados, ha representado un problema que no resuelven estos AG que solo llegan a la simulación. En este sentido para buscar dar una solución se ha utilizado un micro algoritmo genético (µAG) con un número reducido de individuos después de realizar experimentos comparativos con el AG. Utilizar un µAG permite encontrar en un menor tiempo las soluciones y genera una marcha autónoma sobre un robot diseñado con una arquitectura multicore de bajo consumo de energía, tamaño y peso reducido. Los resultados no se quedan en el análisis de una simulación se ha implementado en un sistema embebido (SE) por tanto se presenta el resultado utilizando µAG como un tipo de solución para ayudar en la autonomía de un robot y en aplicaciones como en la exploración de zonas hostiles y de riesgo para el ser humano e incluso en los robots manipuladores.**

I. Introducción

Los AG de acuerdo a con Araujo [1] al igual que otros algoritmos evolutivos son poblacionales. Generan una población de soluciones selecciona las mejores soluciones de acuerdo a la función *fitness*, remplaza la población original generando nuevas soluciones de mejores características como menciona Holland [2]. De esta manera las soluciones robustas se propagan de generación en generación de acuerdo a Golberg [3] tal como se muestra en la figura1.

Figura1. Ciclo de un AG



En las últimas décadas los investigadores han desarrollado un interés particular en los AG para aplicarlos en resolver diversos problemas duros y encontrar con ellos soluciones a los diversos problemas para las diferentes áreas de las ciencias e ingeniería. A continuación se citan algunas investigaciones relacionadas: Gary Parker muestra en su investigación [4], un AG Cíclico que mejora la marcha de un robot cuadrúpedo muestra simulaciones e indica que se ha probado en un robot particular sin mencionar más información de su implementación y de los recursos de cómputo que utiliza. El AG de Chernova [5] es más robusto en los parámetros de evaluación y evita la convergencia prematura mejorando la marcha un 20% del robot AIBO. Gumiel en su investigación [6] se enfoca a estas técnicas evolutivas para mejorar el desempeño del mismo robot. Suzuki y Nishi en su artículo [7] aplican un AG en el robot para mejorar la trayectoria de las patas al dirigir eficientemente la fuerza aplicada e imitar la marcha de un perro. El robot AIBO se ha dejado de fabricar a partir del 2006 el cuál permitía realizar ensayos. La característica primordial de las investigaciones anteriores corresponden a que el robot tiene 16 grados de libertad (GDL). Por otra parte en los robots los manipuladores que tienen menos GDL se aplican las

F. A. Chávez, Alumno de Doctorado del Centro de Innovación y Desarrollo Tecnológico en Cómputo CIDETEC, Instituto Politécnico Nacional, México falexchavez@hotmail.es; fachavez@ittla.edu.mx
J. C. Herrera Centro de Innovación y Desarrollo Tecnológico en Cómputo CIDETEC, Instituto Politécnico Nacional, México jlozada@ipn.mx
J. Sandoval Universidad Autónoma Metropolitana Unidad Lerma, México jsandoval@correo.ler.uam.mx
M. I. Cervantes Instituto Tecnológico de Tlalnepantla isabel_103@yahoo.com.mx; micervantes@ittla.edu.mx

técnicas evolutivas, Gonzalez, Vaca y Peña resuelven la cinemática inversa de un robot manipulador serial de 4 GDL aplicando un AG a partir del modelo cinemático directo screws [8] en un ambiente simulado. Estas técnicas se destacan por su amplia gama de aplicación son robustas se aplican cuando no existen técnicas especializadas no requieren conocimiento del problema y mejoran las técnicas tradicionales. A diferencia de lo anterior el trabajo muestra una investigación en el desempeño de los diferentes AG para la optimización de marcha en robots como es el caso de Chaohong y Hong en sus investigaciones [9] abordan la optimización de marcha con cuatro diferentes AG y solo compara las soluciones encontradas por estos métodos evaluando la velocidad, estabilidad y flexibilidad, mostrando resultados hasta la simulación. Considerando las fortalezas de estos algoritmos se decide utilizar como base un AG y se toma de referencia los µAG como expone Herrera-Lozada en su investigación [10]. Las características que muestran los µAG con poblaciones menores a una decena de individuos encuentran las soluciones de los sistemas utilizando algoritmos genéticos. La implementación de ello en sistemas embebidos focaliza el interés de la presente investigación diseñar micro algoritmos aplicables a la autonomía de marcha en robots que se ejecuten en un SE y no en una computadora. Desarrollar un robot prototipo que beneficie al usuario en realizar investigaciones de marcha o aplicarlo en la exploración en zonas de riesgo para el ser humano y particularmente aplicar los µAG para resolver problemas como determinar la posición final de: brazos manipuladores [8], robots bípedos [11] [12], cuadrúpedos [9] y hexápodos [13].

El artículo se ha organizado como se indica: I Introducción, II Marcha de robots, III Algoritmos genéticos, IV El micro algoritmo genético (µAG), V Implementación en el SE, VI Resultados y discusión, VII Conclusiones.

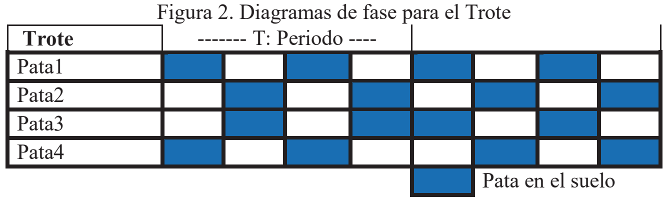
II. MARCHA DE ROBOTS

Para desarrollar la solución de la marcha del robot se realizó el análisis de la cinemática tipo mamífero con la finalidad de considerar las características posibles en el diseño del algoritmo y en el diseño de la plataforma de un robot experimental.

A. Tipo de marcha

La locomoción de los robots con cuatro patas imitan el modo de caminar de los animales como indica Elijah [14], la imitación es de dos tipos:
En el tipo reptil, el cuerpo es más próximo al terreno, su estabilidad es mayor, el consumo de energía es mayor al mamífero por el peso del robot, los torques en las articulaciones son mayores al mamífero y el desplazamiento es menor.
En el tipo mamífero, el cuerpo es más alto por tanto el PDA es más estrecho y el centro de gravedad es más alto, la estabilidad del robot es menor y el consumo de energía depende de la cantidad de eslabones habilitados al mismo tiempo. Se elige esta locomoción para el desarrollo del prototipo con el patrón de mantener dos patas en el suelo y dos patas desplazándose a esto se le denomina trote. En la figura 2 se muestra el patrón de marcha.

Figura 2. Diagramas de fase para el Trote

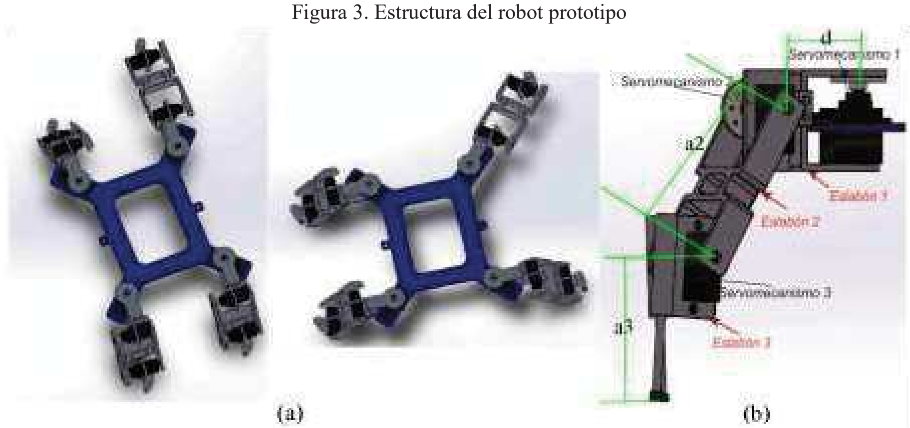


ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

B. Estructura del robot

En la segunda etapa, se ha diseñado y fabricado la estructura para que el robot pueda caminar como un mamífero y reptil. El diseño de cada una de las piezas se hizo en Solid Works y se fabricaron en una impresora 3D. La figura 3, presenta una vista superior del robot diseñado y las características de la estructura.

Figura 3. Estructura del robot prototipo



La estructura básica de diseñado que se presenta en la investigación la constituyen el cuerpo y cuatro patas cada pata tiene tres eslabones. Los eslabones se encuentran unidos a los servomotores como se muestra en la figura 3b. El cuerpo es una plataforma rectangular que permite la instalación de las cuatro patas y los dispositivos electrónicos requeridos para el control. Las patas están dispuestas simétricamente en cada esquina para localizar el CDG dentro del PDA y lograr que los esfuerzos realizados por cada pata se distribuyan uniformemente y así controlar el consumo de corriente.

C. Cinemática del sistema

La tercera etapa consistió en determinar los movimientos angulares de los servomotores que están limitados por la estructura mecánica y forma de los eslabones. El rango de movilidad permite determinar la longitud de desplazamiento la altura máxima y mínima del paso del robot en una superficie horizontal. La altura de paso determina la altura máxima de los objetos que el robot podrá superar en un paso normal. Estos ángulos permiten una diversidad de configuraciones que puede adoptar el robot en la locomoción (movimientos restringidos y no restringidos). Se requiere convertir los desplazamientos angulares de los servomotores a desplazamientos lineales del robot. El análisis cinemático del robot es a través de matrices de transformación homogénea que describen los movimientos así como la geometría espacial, la rotación y traslación de robots con patas. La matriz de transformación homogénea modifica a un vector de posición descrito en coordenadas homogéneas de un sistema de coordenadas de referencia a otro sistema de coordenadas como describe Vukobratovic y Kircanski [15]. La matriz de transformación homogénea contiene cuatro sub matrices:

$$T = \begin{bmatrix} R_{3x3}\ Matriz\ de\ rotación & P_{3x1} Vector\ de\ posición \\ f_{1x3}\ Transformación\ de\ perspectiva & 1x1\ Escalonado \end{bmatrix} \quad (1)$$

La sub matriz superior izquierda indica la rotación del sistema; la sub matriz superior derecha indica el vector de posición del origen del sistema de coordenadas rotado con respecto al sistema de referencia; la sub matriz inferior izquierda indica la transformación de perspectiva y el cuarto elemento diagonal es el factor de escala global. En aplicaciones de robótica, este factor de escala es igual a 1. Se determina la

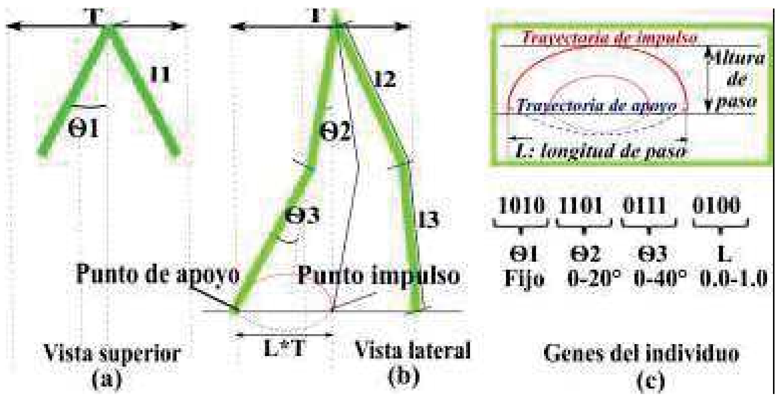
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

matriz homogénea ${}^{0}_{i}T$ para cada articulación lo que indica la localización del i-ésimo sistema de coordenadas de cada articulación con respecto al sistema de coordenadas de referencia y está determinado por la ecuación (2).

$$ {}^{0}_{i}T = \prod_{j=1}^{i^{j-1}} A_j \qquad (2) $$

Al obtener la cinemática de una pata del robot permite encontrar la posición y dirección final de la punta de la pata al caminar a partir de los ángulos de cada articulación de la pata θ_1, θ_2 y θ_3 como se muestra en la figura 4a y 4b.

Figura 4. Relación de ángulos de las una patas del robot



La matriz de transformación ${}^{0}_{i}T$ obtiene dichos puntos p_i y es una función de sus coordenadas homogéneas con i articulaciones con respecto al sistema de referencia establecido para la pata con la ecuación (3) se obtiene dicho punto p_0.

$$ p_0 = {}^{0}_{i}T p_i \qquad (3) $$

Una pata tiene 3 GDL por tanto la ecuación (4) tiene 3 matrices de transformación homogéneas donde:

$$ {}^{0}_{3}T = {}^{0}_{1}A\, {}^{1}_{2}A\, {}^{2}_{3}A \qquad (4) $$

Se realiza la multiplicación de matrices en el orden que se indica dado que el producto de matrices no es conmutativo.
Se utiliza el método Denavit y Hartenberg para determinar la geometría espacial de los elementos de la cadena cinemática (patas del robot) con respecto a un sistema de referencia (centro del robot).
En una cuarta etapa una vez que se tiene la matriz de transformación y se muestra matemáticamente que el desplazamiento del robot lo determinan los valores de los ángulos $\theta_1, \theta_2 .. \theta_i$ de las articulaciones de las patas, se concluye que el AG será quien determine el valor de estos ángulos. En las etapas siguientes se describe a detalle la etapa cinco al aplicar un AG para la marcha autónoma donde se obtiene las características y restricciones de dicho algoritmo y en la etapa seis, se toma de base la etapa anterior para un nuevo diseño implementación y pruebas del µAG para la marcha autónoma.

III. ALGORITMOS GENÉTICOS

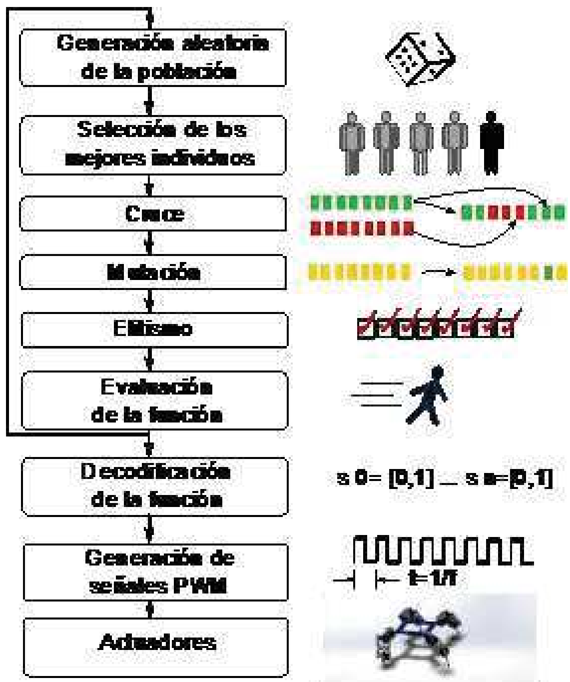
Los AG son procedimientos de búsqueda poblacional se basan en la teoría de la evolución de Darwin, en donde de forma natural se selecciona a los cromosomas o individuos de mejores características genéticas y son soluciones al sistema como describe Arraz y Parra en [16]. Este concepto se aplica en el desarrollo

del AG el cual resuelve problemas con los individuos que encuentra sin tener conocimiento del problema que resuelve. Los pasos del método son:

- Se genera y evalúa la población aleatoria resultado de una función *fitness*.
- Selecciona y permite la reproducción de las mejores soluciones (cromosomas).
- Se genera una nueva población con hijos de mejores características como resultado de la optimización de la función *fitness*, de la cruza, mutación y elitismo. Siempre seleccionando las mejores soluciones o individuos.

Estos pasos se repiten y con estas bases se diseña el AG el cual es el resultado de la función optimizada como exponen Oliveira y Santos [17] en su trabajo. El AG estándar desarrollado sigue el diagrama de flujo de la figura 5.

Figura 5. Diagrama de flujo de implementación de un AG



Con base en la figura 5, los pasos a desarrollar son: la codificación de las soluciones del problema en *n* cromosomas de longitud *L* y dada la función *fitness* o de evaluación proporcionar una medida de referencia para las soluciones que se van generando con el AG. En forma posterior se obtendrá una función resultado de comparar la función *fitness* con las soluciones encontradas y aplicarla al sistema para generar la marcha.

De acuerdo con lo anterior se utiliza un AG simple para el problema de la marcha del robot con las siguientes características:

Codificación del cromosoma: Con los ángulos de rotación de cada articulación se diseña el cromosoma. A cada ángulo se le asigna una cadena de un byte para cada pata se tienen 3 GDL. Obteniendo una cadena de 3 bytes para el cromosoma o individuo solución:

Ángulos de las articulaciones de una pata: $[\Theta_1, \Theta_2, \Theta_3]$.

Las características del cromosoma es una cadena binaria: [01010101 01010101 01010101].

La función *fitness* del sistema son las soluciones o individuos en donde sus genes sean todos unos. [11111111 11111111 11111111].

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Para lograr la condición anterior se define **Las restricciones del sistema al elegir la locomoción tipo mamífero** y se determina en cada articulación los ángulos máximos y el rango de los ángulos válidos $[90°,\ 0° - 40°, 0° - 20°]$ como se muestra en la Tabla I.

Tabla I. ÁNGULOS VÁLIDOS

Reptil		Mamífero	
Articulación	Ángulos válidos	Articulación	Ángulos válidos
Pierna	0 - 30	Pierna	0 - 20
Tobillo	Posición fija	Tobillo	0 - 40
Cadera	0 - 70	Cadera	Posición fija 90

La solución encontrada con el AG es decodificada cada byte corresponde al valor del ángulo de cada articulación. Estos valores se sustituyen en el vector de posición Ecuación 3 y se determina la posición de la punta de la pata en el espacio R^3 esta descripción se muestra en la figura 4c.

IV. EL MICRO ALGORITMO GENÉTICO

El μAG es una versión del AG estándar como ha sido descrito por Goldberg [3] tiene las mismas características descritas en la sección III. Tiene una población reducida de decenas de individuos incluso menos de una decena.

Existe una área de investigación donde aplican AG para la Optimización Multiobjetivos en donde los algoritmos se paralelizan para las soluciones como expone Von Lucken, Hermosilla y Barán [18], sin embargo Herrera en su trabajo [10], utiliza μAG para resolver problemas debe asegurar que estos μAG con una población reducida sean efectivos y eficientes. Lo anterior se logra modificando el AG estándar como muestra el diagrama de flujo de la figura 6.

Figura 6. Diagrama de flujo de μAG



ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Se agrega una convergencia nominal con un ciclo interno que permite encontrar las soluciones con una menor población y una convergencia general con un ciclo externo que permite generar diversidad en las soluciones debido a que el algoritmo se reinicializa después de *i* número de ciclos internos.

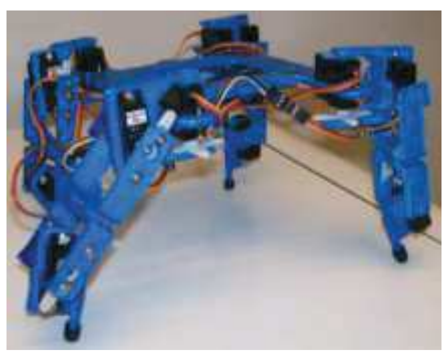
V. IMPLEMENTACIÓN DEL SE

En esta sección se muestran las configuraciones para implementar los algoritmos en el SE y que permita comparar los resultados del desempeño del AG estándar contra el μAG en la marcha del robot.
El SE tiene una arquitectura de 8 cores con 32 puertos entrada/salida, memoria RAM de 32Kbytes, memoria de programa de 32Kbytes u un oscilador de 5 MHz. Tiene acceso directo a periféricos como un CAD, CDA y USB. Se utilizan 12 puertos para conectar en cada uno de ellos un servomecanismo y quedan disponibles 20 puertos del SE para sensores que se puedan requerir en un futuro.
Se configura el computador con la Aplicación para programar el SE. Las aplicaciones se pueden descargar de la página del fabricante (*www.parallax.com*). Se utiliza la configuración para compilar en C se puede programar C++ o ensamblador. La información requerida para manejar el sistema embebido hardware y software se encuentra en la opción de ayuda de la App.

Configurar la App del SE para compilar los algoritmos en lenguaje C.

El robot cuadrúpedo prototipo se ha fabricado con una impresora 3D en material plástico ABS, con las siguientes características.
Cada pata consta de 3 articulaciones con un servomecanismo por pata como resultado el robot tiene 12 GDL. En cada articulación se tiene un servomecanismo figura 7.

Figura 7. Robot cuadrúpedo y SE en el que se implementó el AG y μAG.



VI. RESULTADOS Y DISCUSIÓN

Se han probado y evaluado cuatro algoritmos con las características reportadas en otros trabajos como los expuestos por Gumiel, Sáenz y Quintana [6]. Los resultados obtenidos del AG en la presente investigación son similares a los que ellos obtienen de acuerdo a las tablas que se presentan. Excepto los resultados obtenidos en la investigación con el μAG 5x5 reporta un mejor desempeño con referencia a la media de la generación se incrementa al doble.

A. Desarrollo y experimentos

Los algoritmos deben cumplir las siguientes características:

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

- Individuos de un byte. La función *fitness* es alcanzar el valor de 255.
- Población inicial de 16 individuos generados aleatoriamente.
- Generaciones que van de 10, 100 y 1000 unidades.
- Selección por elitismo de los dos mejores individuos.
- Un cruce en cada byte del individuo
- Mutación de un bit en cada byte.

El algoritmo debe indicar la generación y el tiempo en que encuentra la solución para la marcha. Se grafica el valor *fitness* del mejor individuo y la media de la generación del valor *fitness* de los individuos:

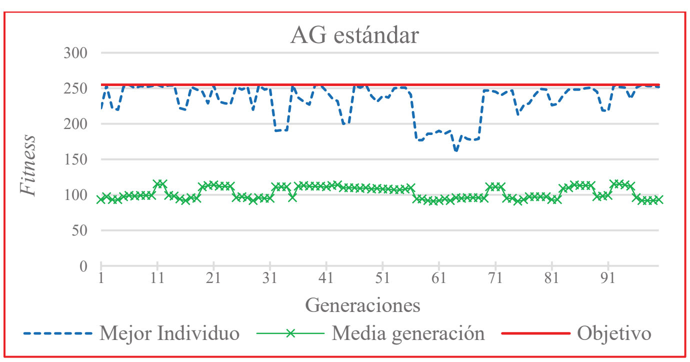
1. Un AG estándar con 1000 generaciones.
2. Un AG estándar de 100 generaciones. El AG estándar se ejecuta en el SE con 100 generaciones.
3. Un µAG con 10 generaciones.
4. Un µAG 5x5 (5 ciclos externos X 5 ciclos internos= 25 generaciones).
5. Comparar el desempeño de los AG y el µAG para implementar cada uno de los algoritmos en el robot prototipo. El robot realiza la marcha y recorre una distancia de 0.9 m. Se mide el tiempo en que recorre dicha distancia.
6. Se realiza la prueba en 10 ocasiones para cada algoritmo y se registran los tiempos medios en una tabla.

B. Resultados

1. El AG estándar con 1000 generaciones no se ejecuta en el SE. Se identifica que el algoritmo tiene un tamaño de 29 Kbytes y durante la ejecución se realizan dos poblaciones de 1000 individuos el resultado corresponde a rebasar la capacidad de memoria de programa del SE. Al reducir las poblaciones a 200 individuos el algoritmo se puede ejecutar en el SE debido a que su tamaño se reduce a 26 Kbytes.

2. El AG de 100 generaciones se ejecuta en el robot se muestran resultados en la figura 8.

Figura 8. AG estándar

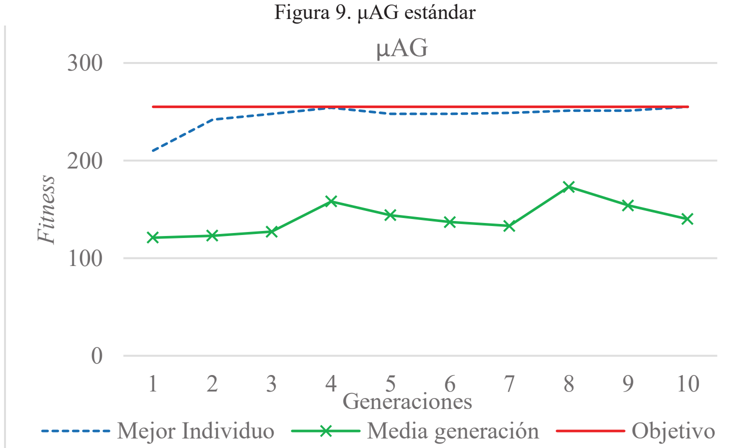


Con estos datos el algoritmo encuentra las soluciones con una media de las generaciones de 100 unidades.

3. El µAG con 10 generaciones encuentra la solución en la generación 4. La media de las generaciones es mayor al AG y tiene un valor de 130 unidades se muestra en la figura 9.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

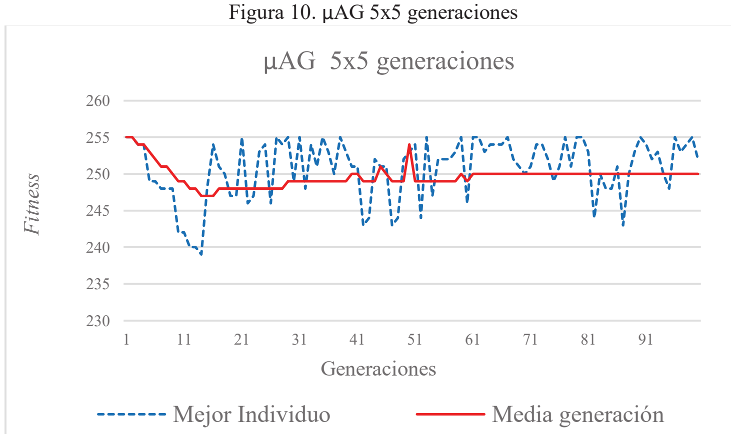
Figura 9. µAG estándar



El espacio en memoria del programa no es mayor a 24K bytes. El algoritmo programado tiene un ciclo interno de convergencia nominal de 5 iteraciones y un ciclo externo de convergencia general, en total este algoritmo ejecuta 5 generaciones en cada corrida y en promedio de dos corridas encuentra la solución del sistema. La media de las generaciones en los µAG es mayor respecto al AG.

4. El µAG 5x5 = 25 generaciones encuentra en menor tiempo y en menos generaciones las soluciones del sistema. El algoritmo contiene las mejores soluciones encontradas con el manejo de los operadores genéticos: elitismo de los dos mejores individuos y una doble mutación en las nuevas generaciones; estas características favorecen el desempeño del algoritmo para encontrar la solución, para este algoritmo la media de las generaciones se aproxima al 98% a la solución se muestra en la figura 10.

Figura 10. µAG 5x5 generaciones



5. Se realizaron pruebas al robot cuadrúpedo prototipo a diferentes velocidades (ver columna Velocidad (m/s)). Se determina el tiempo (ver en la columna Tiempo (s)) en que se desplazó una distancia de 90 cm, se muestran resultados en la Tabla II.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Tabla II. TIEMPOS DE DESPLAZAMIENTO (a) µAG y (b) AG

Robot Mamífero distancia recorrida 90 cm con µAG		
Tiempo (s)	Velocidad (m/s)	Comentarios
27.8	0.032	Semi estable
25.7	0.035	Semi estable
24.5	0.037	Semi estable
21.3	**0.042**	**Estable y recto**
17.8	**0.051**	**Estable y recto**
15.2	**0.059**	**Estable y recto**
12.4	0.073	Estable, no recto
10.1	0.089	Estable, no recto
9.6	0.094	Estable, no recto
9.5	0.095	**Rápido riesgo al sistema**
(a)		Valor óptimo

Robot Mamífero distancia recorrida 90 cm con AG		
Tiempo (s)	Velocidad (m/s)	Comentarios
88.7	0.007	Estable, lento
80.9	0.007	Estable, lento
73.1	0.008	Estable, lento
65.3	0.009	Estable, lento
58.3	0.01	Estable, lento
47.9	0.013	Estable y recto
42.8	0.014	Estable y recto
31.8	**0.019**	**Estable, recto**
21.4	**0.028**	**Estable, recto**
16.3	0.037	Semi estable
(b)		Valor óptimo

Los avances obtenidos permitieron observar que el µAG tiene un mayor desempeño ya que en un menor tiempo encuentra las soluciones para la marcha del robot con respecto al AG y con estas condiciones la marcha del robot es continua. En este último experimento se ejecutan 10 veces los AG y el µAG 5x5 y en cada ejecución se obtienen los tiempos en que cada algoritmo encuentran las soluciones para y se muestran los tiempos en la Tabla III.

Tabla III. TIEMPOS DE EJECUCIÓN AG vs µAG 5x5

Tiempos No	AG Generación	Tiempo (s)	µAG 5x5 Ciclo	Tiempo (s)
1	21	0.143	4	0.147
2	15	0.101	1	0.008
3	6	0.033	3	0.118
4	56	0.32	2	0.06
5	68	0.329	2	0.06
6	10	0.067	1	0.008
7	7	0.02	1	0.008
8	5	0.014	1	0.008
9	10	0.068	2	0.06
10	87	0.483	2	0.06
Valor medio	28.5	0.1578	1.9X5= 9.5	0.0537

En esta aplicación la función optimizada es una cadena de bits decodificada y almacenada en variables independientes. Cada variable proporciona la información en cada PWM determinando con ello el valor de la variable para el cálculo del ángulo máximo de la articulación que corresponde al ciclo de trabajo de la señal PWM para generar la marcha del robot.

C. Discusión de Resultados

En el trabajo se ha logrado una correcta implementación del AG con una poblaciones de 16 individuos y generaciones de 200 lo que permiten generar la marcha en un robot cuadrúpedo implementado en un SE. En la segunda etapa el µAG en el SE con una población máxima de 16 individuos y 10 generaciones se encuentra que el programa corre y genera una marcha continua en el robot. La característica primordial encontrada es que generan poblaciones reducidas y en el caso del µAG se identifica que una decena es suficiente para encontrar las soluciones de los sistemas. Trabajar estas características permite encontrar las soluciones válidas en un menor tiempo y genera un patrón de marcha.

VII. CONCLUSIONES

En la presente investigación se genera un acercamiento para una correcta implementación del μAG en la plataforma diseñada: SE-robot cuadrúpedo prototipo. Además la plataforma puede auxiliar o contribuir para evaluar diversos AG de población reducida menores a 200 generaciones. Es la base para diseñar μAG lograr un aprendizaje Bio-inspirado que permita generar la marcha en un robot cuadrúpedo y se puede implementar a robots de mayor número de patas e incluso robots manipuladores. Una decena de cromosomas o individuos es suficiente para encontrar las soluciones de este tipo de sistemas estas características permite encontrar las soluciones válidas en un menor tiempo y generar un patrón de marcha.

El presente trabajo también proporciona una guía de implementación de un μAG o AG de población reducida en un SE; demanda menos recursos de procesamiento sin la necesidad de usar una computadora. Se puede someter a consideración como un prototipo educativo. Dotar a este tipo de sistemas de baterías y un SE les permite una mayor autonomía.

Los μAG son una alternativa para resolver problemas en donde los GDL crecen como es el caso de los robots con patas y robots manipuladores. Para estos sistemas el grado de complejidad aumenta al buscar soluciones tradicionales por medio de matrices además crece la demanda de recursos de cómputo.

APLICACIONES

Lo anterior permite mejorar la autonomía del robot y enfocar los robots en la aplicación de marcha con obstáculos así como marcha en regiones con diferentes superficies, exploración planetaria, agricultura, zonas de riesgos para el manejo de explosivos, zonas con ambientes peligrosos y contaminados, robots manipuladores aplicados a la industria, entre otros.

AGRADECIMIENTOS

Los autores agradecen a la Secretaría de Investigación y Posgrado del IPN por el soporte económico a través del Proyecto SIP 20160019 y al CONACyT por la beca para estudios de doctorado.

REFERENCIAS

[1] Araujo, L. "Algoritmos evolutivos: un enfoque práctico". *Alfaomega.* 2009.

[2] Holland, J. H. "Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence". *MIT press.* 1992.

[3] Goldberg, D. E. "Genetic Algorithms in search Optimization e Machine Learning". *USA: Addison-Wesley Publishing Comapny Inc.* 1989.

[4] Parker, G. B. "Using Cyclic Genetic Algorithms to Learn Gaits for an Actual Quadruped". *IEEE International Conference on systems*. 2011. p1938--1943. DOI: 10.1109/ICSMC.2011.6083871.

[5] Chernova, S. " An evolutionary approach to gait learning for four-legged robot". *Intelligent Robots and Systems. Proceedings. 2004 IEEE/RSJ International Conference*. 2004. 2562--2567. vol 3. DOI: 10.1109/IROS.2004.1389794.

[6] Gumiel-Moreno Pablo, S.-A. Y.-M. "Implementación de técnicas de computación evolutiva a la programación automática de un robot". *Madrid España: Universidad Carlos III de Madrid*. 2009. Obtenido de http://hdl.handle.net/10016/6581

[7] Suzuki, H." Animal Gait Generation for Quadrupedal Robot". *Innovative Computing, Information and Control, 2007. ICICIC'07.*

Second International Conference on IEEE, 2007. DOI:10.1109/ICICIC.2007.169.

[8] Vaca-González J J, Peña-Caro C A y Vacca- Gonzalez H. "Cinemática inversa de robot serial utilizando algoritmo genético basado en MCDS". *Tecnura, Bogotá,* 2015. Vol. 19(44), p.33-45. doi:dx.doi.org/10.14483/udistrital.jour.tecnura.2015.2.a02.

[9] Cai, C. "Performance Comparisons of Evolutionary Algorithms for Walking Gait Optimization". *International Conference on(IEEE).* 2013. p. 129-134.DOI: 10.1109/ISCC-C.2013.100.

[10] Herrera-Lozada JC. "Propuesta de una Metodología Generalizada para. CICOS 2009". *Congreso Internacional de Cómputo en Optimización y Software, Memorias del 7mo Congreso de Cómputo CICO.* 17-20 Noviembre 2009, UAEM, México, p.189-199.

[11] Dip, G. "Genetic algorithm-based optimal bipedal walking gait synthesis considering tradeo between stability margin and speed". *Cambridge Univ Press(Robotica)*. Vol. 27/03, p. 355-365. DOI:10.1017/S026357470800475X.

[12] Barroso F, G. J. "Optimización Evolutiva de la Locomoción de un Robot Bípedo". *Conference DRT4 ALL Universidad de Huelva España*. 2011. Vol.1, p. 46-60.

[13] Gonzalez de Santos P, G. E. (2012). "Minimizing energy consuption in hexapod robots". *Advanced Robotics.* 2012. Vol. 23/6, p.681-704. DOI:10.1163/156855309X431677

[14] Elijah-McKenzie J. "Design of Robotic Quadruped legs". *Massachusetts, USA: Massachusetts Institute of Technology.* 2012. DOI:785729031-MIT

[15] Vukobratovic M and Kircanski. "Kinematics and Trajectory Synthesis of Manipulation Robots". *Berlin Heidelberg: Springer-Verlag*. 2013. DOI:10.1007/978-3-642-82195-0

[16] Arraz de la Peña, J. "Parallel Genetic Algorithm for Creation of Sort Algorithms". *Springer Berlin Heidelberg, 8083(Computational Collective Intelligence. Technologies and Applications)*. 2013. p. 367-376. DOI:10.1007/978-3-642-40495-5_37.

[17] Oliveira M, S. C. "Multi-objective parameter CPG optimization for gait generation of a quadruped robot considering behavioral diversity". *IEEE/RSJ International Conference on Intelligent Robots and Systems.* 2011. p. 2286-2291. DOI:10.1109/IROS.2011.6094819.

[18] Von Lucken, C. “Algoritmos Evolutivos para Optimización Multiobjetivo: un estudio comparativo en un ambiente paralelo asíncrono”. *X Congreso Argentino de Ciencias de la Computación*. 2004

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 13, páginas 133 - 139.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Sistema prototipo de adquisición de imágenes estereoscópicas

J. A. Olvera-Balderas, J. C. Sosa-Savedra y M. A. Oloño García

***Resumen* — Los sistemas de visión estereoscópica son aquellos que poseen características para la obtención de la información tridimensional de un ambiente. Existen diversas áreas de aplicación como sistemas para la navegación robótica, reconstrucción de objetos tridimensionales, manipulación de brazos robóticos, entre otras. En este trabajo se presenta el desarrollo de un sistema prototipo capaz de obtener un par de imágenes estereoscópicas con la ayuda de dos cámaras y empleando tecnología reconfigurable. Estas imágenes serán procesadas para obtener como producto final una imagen estereoscópica anáglifa.**

I. INTRODUCTION

Gracias a la visión binocular, el ser humano posee la capacidad de ver el entorno en tres dimensiones, lo que significa poder visualizar dos imágenes con diferentes perspectivas, una para cada ojo. Este hecho presenta grandes ventajas, entre las que destacan: el captar la ubicación de objetos y la percepción de profundidad del entorno que nos rodea [1]. Por ejemplo: el ser humano utilizaba esta habilidad para cazar, conociendo perfectamente la distancia a la que se hallaba un animal, arrojaban sus lanzas contra él.

La visión estereoscópica se encuentra basada en la visión binocular. Existen diversas técnicas de estereoscopía, pero en general, estas consisten en presentar imágenes para cada ojo con distintas perspectivas, con el fin de crear al sujeto una sensación de una tercera dimensión (3D) [2].

En la actualidad, la reproducción de imágenes se realiza a través de televisores, monitores de computadora, proyectores, dispositivos móviles, entre otros. Sin embargo, a diferencia de la visión binocular, la mayoría de estos dispositivos únicamente posee la capacidad de mostrar imágenes con una sola perspectiva, provocando la pérdida de aquella tercera dimensión. En la rama de la robótica, los robots tele-operados presentan grandes problemas al no tener un sistema de visión binocular [3]. No obstante, con los grandes avances tecnológicos se ha desarrollado por ejemplo televisores 3D, los cuales, implementan técnicas de visión estereoscópicas como la activa y polarización [4].

Un problema reciente es que esta tecnología tiene un alto costo de adquisición. La mayoría de los sistemas desarrollados en instituciones que tienen relación con la visión estereoscópica, emplean como elemento de ejecución a un computador de propósito general y son implementados en lenguajes de alto nivel como java [2] o .NET [5]; implicando una baja portabilidad del sistema.

Por tal razón, en este trabajo se propone el desarrollo de un sistema embebido prototipo de visión estereoscópica, con las ventajas de ser de bajo costo y con una alta portabilidad. Para ello se empleará tecnología lógica reconfigurable y dos sensores de visión CMOS. Cada sensor tiene la capacidad de adquirir imágenes de 1.3 MP a 30 fps.

II. ANÁLISIS DEL SISTEMA

A. *Descripción*

Se requiere de un sistema de visión que implemente la técnica de visualización estereoscópica anáglifa [2]. Este debe contar con la capacidad de obtener dos imágenes estereas simultáneamente, a través de dos cámaras, y con ellas lograr la reproducción de una imagen estereoscópica tipo anáglifo, la cual, será transmitida mediante el protocolo VGA a un monitor convencional, de esta forma, el usuario podrá visualizar la imagen con ayuda de unas gafas de filtro anaglífico.

B. *Técnica de visualización estereoscópica pasiva de anáglifo.*

La técnica de anáglifo a diferencia de la estereoscopia activa, requiere de unas gafas de bajo costo y de fácil acceso. Esta técnica se encuentra basada en el filtrado de las ondas de longitud de la luz visible. Pueden implementarse técnicas de filtrados de colores aditivos o sustractivo [6]. El monitor debe presentar ambas imágenes (la imagen con perspectiva del ojo izquierdo y la del derecho) superpuestas. Sin embargo, antes de ser superpuestas, en cada imagen se debe aplicar un filtrado de color ya sea aditivo o sustractivo. Cuando el monitor muestra ambas imágenes superpuestas, la luz que intenta pasar a través de cada lente

J.A. Olvera-Balderas, estudiante del Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada, Querétaro, 76090, México. (Cel: 442-579-9991; e-mail: armando.ob@hotmail.com).

J.C. Sosa-Savedra, profesor-investigador del Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada, Querétaro, 76090, México. (e-mail: jcsosa@ipn.mx)

M.A. Oloó-García estudiante en la Escuela Superior de Computo del Instituto Politécnico Nacional, Ciudad de México, 07738, México.

es filtrada, permitiendo el paso de sólo ciertas longitudes de onda (color), i.e. permite separar nuevamente las imágenes superpuestas, lo que ayuda a entregar a cada ojo su imagen correspondiente. Por ejemplo, a la imagen del ojo izquierdo se le aplica un filtrado de color rojo (lo que significa que estamos eliminando sus colores verde y azul [6]), y a la imagen del ojo derecho se le aplica un filtro cian (lo que significa que estamos suprimiendo su color rojo [6]), después, son superpuestas y mostradas por el monitor. Supongamos que los lentes contienen un filtro rojo (lente izquierdo) y uno cian (lente derecho), a la hora en que la luz de la imagen intenta pasar por las gafas, el lente rojo sólo permitirá el paso de la imagen a la que se le aplico el filtro rojo, mientras que el otro lente solo permitirá el paso de la imagen derecha. Entregando así una imagen distinta a cada ojo, consiguiendo así una ilusión 3D.

C. *Hardware.*

Para la implementación hardware, las opciones VLSI (Very Large Scale Integration), ASIC (Application Specific Integrated Circuit) y PLD (Program Logic Device) proporcionan diseños hechos a la medida del sistema deseado y dan una gran velocidad de procesado, sin embargo, los diseños VLSI y ASIC no pueden ser reconfigurados y tiene un alto costo de fabricación, a diferencia del diseño con un PLD. Mientras tanto, un DSP (Digital Signal Processor) presenta una gran flexibilidad de programación, puede emplear lenguajes de alto nivel, sin embargo, al ser tan flexibles, la velocidad de ejecución de cualquier algoritmo computable es más lenta debido a que la arquitectura no está especializada para el algoritmo a ejecutar, otra desventaja es el costo de los mismos respecto a un diseño con un PLD. El punto intermedio, entre un chip hecho a la medida y un chip con una arquitectura ya estructurada, es el empleo de los dispositivos lógicos programables (PLD).

Las ventajas que proporcionan son: 1) la arquitectura de estos puede ser adaptada al algoritmo que se desea implementar mediante un lenguaje descriptivo de hardware (HDL), 2) son muy eficientes al implementar algoritmos que pueden ser ejecutados de forma paralela, 3) son reconfigurables, 4) tienen una alta portabilidad [7]. Cabe mencionar que los lenguajes para descripción de hardware (HDL) más comunes son VHDL y Verilog [8].

Una implementación hardware proporciona una mayor velocidad de procesamiento que una implementación software. Dentro de las alternativas de una implementación hardware, los PLD son la alternativa más viable para la implementación de un sistema de visión [9].

Los PLD pueden ser clasificados en SPLD (Simple Programable Logic Device), CPLD (Complex Programable Logic Device) y FPGA (Field Program Gate Array). Los SPLD aun que son de bajo costo, presentan la desventaja de tener una baja cantidad de macro celdas, por lo tanto, no es posible implementar sistemas robustos; por otro lado, la tecnología de los CPLD y FPGA ha mejorado mucho últimamente, lo que permite implementar sistemas mucho más robustos que en un SPLD y a un bajo consumo eléctrico. Algunas de las aplicaciones más comunes que implementan CPLD son [10],[11]: sistemas para control de encendido, sistemas de configuración, expansiones de entrada/salida, interfaces de conexión, entre otras; en cuanto a los FPGA, son comúnmente utilizados en soluciones más robustas; por ejemplo, aplicaciones industriales [10]. De lo anterior se puede concluir que la mejor opción para implementar este proyecto es la tecnología FPGA.

III. ARQUITECTURA DEL SISTEMA

En la Figura 1 se puede apreciar la arquitectura del sistema en general. En las siguientes secciones se realizará la descripción de cada una de las etapas y módulos que componen el sistema.

A. *Cámaras*

En esta etapa se requiere del uso de dos cámaras, con la característica de que los sensores se encuentren físicamente separados por una distancia aproximada de 45mm a 75mm [2]. De acuerdo a la investigación realizada, las cámaras que serán utilizadas tendrán una separación de 35mm.

B. *Controlador de cámara*

El sistema debe tener la capacidad de configurar, sincronizar y controlar las dos cámaras conectadas a él, con el fin de obtener dos imágenes simultáneas. Para ello, se debe diseñar un "Controlador" o Driver; a continuación, los datos obtenidos por el controlador, son transmitidos al bloque "Reconstrucción de imágenes". El driver debe encargarse de configurar las cámaras de modo que estas brinden imágenes en formato VGA (640(H) x 480(V) pixeles).

Debido a las características de las cámaras, se requiere que éste módulo tenga la capacidad de comunicarse con la cámara a través del protocolo de comunicación I^2C.

C. *Reconstrucción de imágenes RAW a 320 x 240 pixeles RGB*

Este bloque tiene la función de interpretar los datos proporcionados por el controlador de la cámara. A partir de ello, reconstruirá la imagen capturada entregando a la salida imágenes con un tamaño de 320 x 240 pixeles en formato RGB.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

D. Filtros anaglíficos y superposición de imágenes estereoscópicas

La técnica de visualización 3D tipo anáglifo requiere de dos filtrados diferentes (filtro rojo y cian en este caso), los cuales, son aplicados a la imagen correspondiente (imagen izquierda y derecha). Una vez aplicado el filtro, las imágenes deberán de ser superpuestas entre sí.

E. Controlador SRAM

El objetivo de éste módulo es poder almacenar en una memoria SRAM las imágenes estereoscópicas y a su vez, poder leerlas y entregarlas a la salida cuando el siguiente módulo lo requiera.

F. Reconstructor de imágenes a 640 x 480 pixeles

Éste módulo se encarga de redimensionar las imágenes de 320x240 pixeles (almacenadas en la SRAM) a imágenes de un tamaño de 640x480 pixeles.

G. Controlador de salida VGA

Este controlador es el encargado de transmitir la imagen final hacia el monitor. El formato de las imágenes que debe recibir es VGA (640(H) x 480(V) pixeles) con una resolución de 10 bits por color.

H. Unidad de control

Cada uno de estos bloques funcionales es controlado y sincronizado mediante la “Unidad de Control”. Está diseñada especialmente para este sistema. Las funciones que realiza en orden la unidad de control son los siguientes:

1. Activar los controladores de ambas cámaras para comenzar la configuración de estas.

2. Una vez terminada la configuración de ambas cámaras, se indica simultáneamente a los controladores de las cámaras que comiencen la recepción de imágenes; a la vez, son activados los bloques: reconstructor de imágenes RAW a RGB, controlador SRAM, reconstructor de imágenes a 640x480 pixeles y por último, el controlador VGA.

No obstante, cada uno de los bloques tiene la capacidad para detectar el inicio y fin de una imagen, de esta forma al terminar de procesar los datos de una imagen, estos vuelven a su estado inicial para comenzar la recepción de la siguiente imagen.

I. Visualización de imágenes estereoscópicas tipo anáglifo

Finalmente, un monitor se encontrará mostrando las imágenes proporcionadas por el sistema. Con la ayuda de unas gafas con filtro anáglifo (filtro rojo y cian), el usuario tendrá la capacidad de apreciar las imágenes en 3D.

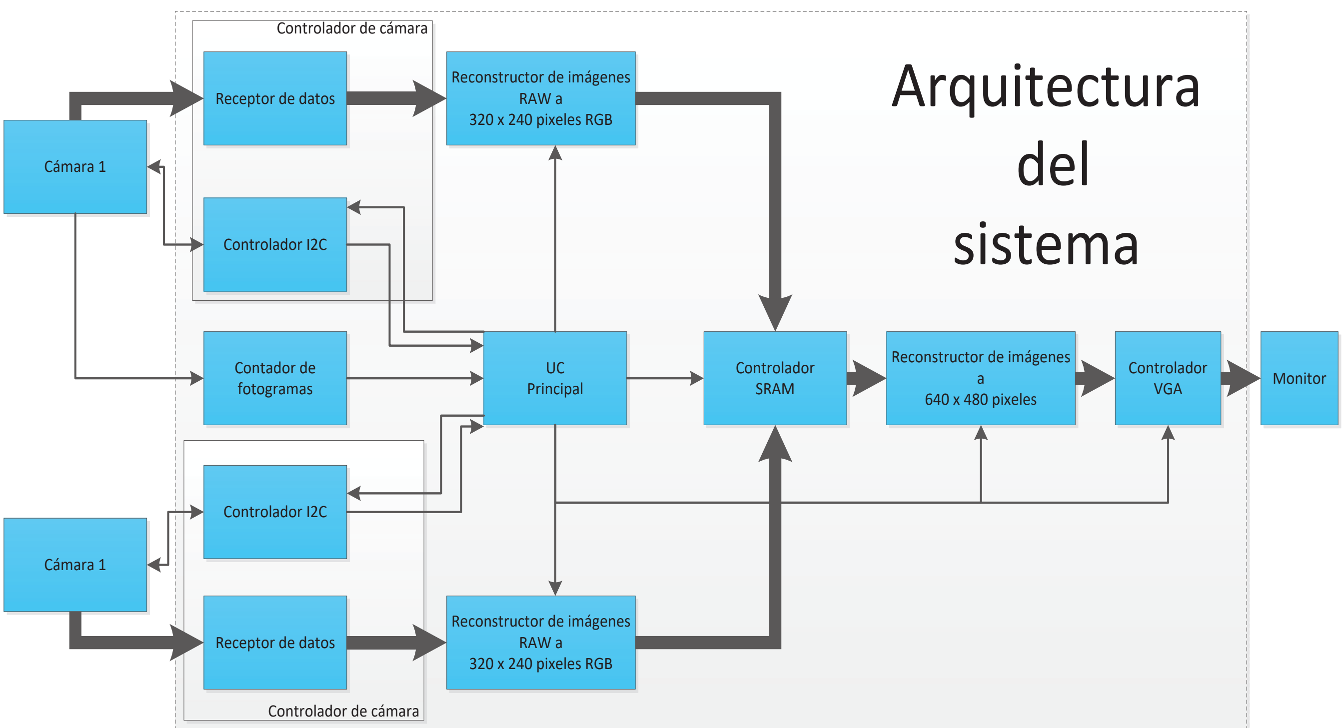


Figura 1. Arquitectura del sistema de adquisición de imágenes estereoscópicas.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

IV. Herramientas Utilizadas

A. *Kit de desarrollo TRDB_DC2*

Cuenta con dos cámaras MT9M011, las cuales contienen un sensor tipo CMOS de 1.3Mega pixeles. También cuenta con un conector de 40 pines, para realizar una conexión satisfactoria con los conectores de expansión integrados en las tarjetas de desarrollo DE2/DE1 y TREX-C1 a través de un cable IDE [12], [13].

B. *Software*

Debido a las tecnologías seleccionadas para la implementación del sistema, el software de desarrollo utilizado fue "Quartus II ® versión 13.01 Services Pack" de Altera.

Reporte de compilación: En la tabla 1 se presenta el reporte general del código del sistema, donde puede observarse que se requirió menos de la mitad de recursos de la tarjeta de desarrollo para la implementación del sistema, aunque en cuanto a entradas y salidas físicas se requieren demasiados pines, que podría ser una desventaja en caso de tener un FPGA con menos entradas y salidas disponibles.

Tabla 1. Resultados de compilación.

Descripción	**Resultado**
Flow Status	Successful
Quartus II 64-Bit Version	13.0.1 Build 232 06/12/2013 SP 1 SJ Web Edition
Revision Name	DE2_CCD
Top-level Entity Name	DE2_CCD
Family	Cyclone II
Device	EP2C35F672C6
Total logic elements	1,683 / 33,216 (5 %)
Total combinational functions	1,455 / 33,216 (4 %)
Dedicated logic registers	1,023 / 33,216 (3 %)
Total registers	1023
Total pins	425 / 475 (89 %)
Total memory bits	138,240 / 483,840 (29 %)
Embedded Multiplier 9-bit elements	0 / 70 (0 %)
Total PLLs	0 / 4 (0 %)

V. Resultados y Conclusiones

El inicio del sistema es controlado a través de la activación de un switch de la tarjeta de desarrollo. Una vez activado, la transmisión de las imágenes estereoscópicas se realiza instantáneamente.

Se logró la adecuación del sistema para que entregue a la salida 30 fps, la mayor velocidad que se logró alcanzar para imágenes con una resolución de 640x480 pixeles, con una frecuencia de reloj de 25 MHz en el pin MCLK de las cámaras (Frecuencia máxima recomendada por el fabricante).

Se realizó la toma de la figura de un caballo de yeso (Figura 2), a la que se le aplicó la estereoscopía anáglifa para la prueba del sistema.



Figura 2. Figura de caballo de yeso, forma original.

En la Figura 3, se puede apreciar la imagen 3D final proporcionada por el sistema, en ella se pueden apreciar dos imágenes estereoscópicas superpuestas, una correspondiente al filtro rojo y otra al cian.



Figura 3. Imagen estereoscópica de tipo anaglífica del caballo de yeso.

En las figuras 4 y 5 se muestran las imágenes estereoscópicas por separado. Cabe mencionar que para obtener estas imágenes, se requiere bloquear completamente un sensor a la vez.



Figura 4. Imagen del caballo de yeso con filtro anaglífico Cian.



Figura 5. Imagen del caballo de yeso con filtro anaglífico Rojo.

El procesamiento de estas imágenes y su apreciación en 3D requiere de unos lentes anáglifos (Figura 6), lo cual puede presentar algunos inconvenientes para el usuario.



Figura 6. Lentes anáglifos con filtro cian y rojo.

A pesar de lograr la estereoscopía anaglífica, la capacidad de trabajo de los sensores, se está quedando corta en cuanto a procesamiento de imágenes hoy día que se manejan resoluciones superiores a los 5MP, aunque se puede mejorar la velocidad de fotogramas por segundo, esto no se puede probar a mayores resoluciones, aunque una mayor velocidad puede ayudar en trabajos de análisis de movimiento.

Por otro lado, el uso de lentes anáglifos también está quedando obsoleto y se está remplazando por técnicas diferentes, ya que el anáglifo presenta dificultades para el usuario como lo son el poder enfocar la imagen en 3D, y algunas personas presentan un efecto de vista cansada a los pocos minutos de utilizar las gafas debido a los colores que el ojo percibe; presentando también la desventaja de que al ser imágenes en base a filtros de 2 colores, se pierden colores percibidos por el ojo humano.

A trabajo futuro, existe la posibilidad de trabajar con resoluciones mayores para tener una mejor calidad de imagen, aunque por el hecho de que este tipo de procesamiento de imágenes 3D está quedando obsoleto, sería conveniente trabajar con el procesamiento 3D SBS (*Side By Side*).

REFERENCIAS

[1] K. Rogers, The Eye The Physiology of human perception, New York: Britannica Educational Publishing, 2011.
[2] I. N. R. García, Sistema de visión estereoscópica basado en anaglifo para aplicaciones de realidad virtual, Centro de Innovación y Desarrollo Tecnológico en Computo, Mexico, D.F., Julio 2008.
[3] J. C. M. R. G. F. R. D. J. S.-H. F. J. Jiménez, Sistema de visión de apoyo a la navegación de un robot móvil en invernaderos, Universidad de Almería.
[4] SONY, «SONY,» 2014. [En línea]. Available: www.sony.com. [Último acceso: 2014].
[5] R. R. G. J. A. R. S. Daniel Martín Carabias, Sistema de visión estereoscópica para la Navegación autónoma de vehículos no tripulados., Universidad Complutense de Madrid Facultad de Informática, 2010.
[6] M. Langford, Manual de Laboratorio Fotográfico, Madrid: TURSEN-HERMANN BLUME, 1994.
[7] V. H. G. Ortega, «Dropbox (SPLD),» 2013. [En línea]. Available: https://www.dropbox.com/sh/vo66hnsfo7fo3u2/AABnjhaRdFNAdi5yskqEQeKZa/Dise%C3%B1o%20de%20Sistemas%20Digitales/DOCUMENTOS/Cap%C3%ADtulo_1%20Intro%26SPLD.pdf?dl=0. [Último acceso: 10 2014].
[8] J. A. B. G. Fernando Pardo Carpio, VHDL Lenguaje para síntesis y modelado de circuitos, MADRID, España: Ra-Ma, 5 de diciembre de 2003.

[9] J. C. S. Savedra, Sistema de visión basado en procesado guiado por cambios y lógica reconfigurable para el análisis de movimiento de alta velocidad, UNIVERSITAT DE VALÉNCIA, 28 de junio de 2007.
[10] Altera, «Altera Measurable Advantage,» [En línea]. Available: http://www.altera.com/devices/dvcs-index.html. [Último acceso: 10 2014].
[11] Xilinx, «Xilinx,» 2014. [En línea]. Available: http://www.xilinx.com/. [Último acceso: 2014].
[12] Terasic, «Terasic,» 2013. [En línea]. Available: http://www.terasic.com.tw/cgi-bin/page/archive.pl?Language=English&CategoryNo=68&No=50&PartNo=2. [Último acceso: 2014].
[13] Terasic, «Terasic,» 2013. [En línea]. Available: http://www.terasic.com.tw/cgi-bin/page/archive.pl?Language=English&CategoryNo=68&No=281&PartNo=2. [Último acceso: 2014].

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 14, páginas 140 - 146.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Desarrollo de un sistema para comunicación utilizando EOG*

Miguel I. Ceballos Pérez[1], Eduardo Morales Sánchez[1]

[1]Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada, Instituto Politécnico Nacional

***Resumen* — En México hay casi medio millón de personas con limitación para hablar o comunicarse, entre las cuales algunas no pueden siquiera moverse. El propósito de este trabajo es presentar una alternativa que sea rápida, eficiente y sencilla, para que personas con limitación para hablar y moverse puedan comunicarse con otras personas de manera verbal.**

El sistema utiliza la actividad eléctrica generada por los músculos implicados en el parpadeo de los ojos, para la selección de frases de necesidad básica, el sistema está basado en un circuito integrado de Texas Instruments (ADS1299) y el software Labview, en este trabajo se presenta la metodología que utilizada para desarrollar este sistema.

Palabras claves – Electroculografía, Adquisición de señales, actividad muscular

I. INTRODUCCIÓN

"El habla es natural para el ser humano pero también es parte de la necesidad social del ser humano expresado a través de las palabras" [1], sin embargo, hay enfermedades que disminuyen la capacidad de comunicación de las personas, al menos en México hay más de 477,000 personas con limitaciones para hablar o comunicarse [2]. Se han inventado técnicas y métodos para que estas personas logren comunicarse, como lenguaje de señas o dispositivos eléctricos para personas mudas, hay personas que tienen limitación para comunicarse y no logran controlar las partes de su cuerpo por completo.

Los músculos que menos daños presentan son los músculos oculares, por lo que resulta atractivo utilizar el movimiento ocular para desarrollar tecnología que permita a personas con ciertos tipos de discapacidad a comunicarse con otras personas. Se han desarrollado diferentes sistemas de comunicación utilizando el movimiento ocular, sin embargo, una gran parte de estos se basa en imágenes, estos sistemas son muy precisos, pero necesitan de sistemas costosos y un alto procesamiento.

El electrooculograma (EOG) es un método que puede detectar los movimientos oculares, el cual mide el potencial eléctrico entre la córnea (eléctricamente positiva) y la parte posterior del ojo (eléctricamente negativa), por esta razón es que el globo ocular se considera como un dipolo eléctrico, de este modo, el movimiento del globo ocular provoca cambios en la dirección del vector eléctrico del dipolo eléctrico. [3]

Las mediciones electrooculográficas se realizan con el uso de amplificadores en modo diferencial; existen dos formas de medir una señal electrooculográfica, la primera es midiendo la actividad eléctrica de forma horizontal y la segunda es midiendo la actividad eléctrica de forma vertical, como se muestra en la Figura 1, aunque la principal razón de realizar un electrooculograma es registrar la actividad eléctrica del movimiento del ojo, el electrooculograma es capaz de medir la energía eléctrica producida por los músculos cuando se realiza un parpadeo o se cierra un ojo o los dos.

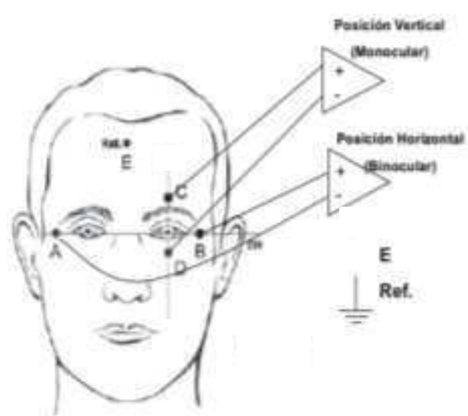


Figura 1 Diagrama de colocación de electrodos para EOG

*Investigación respaldada por el IPN.

Miguel Iván Ceballos Pérez, estudiante de maestría del Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada del IPN unidad Querétaro, México, (teléfono 442-570-9632; e-mail: miguelivan.ceballos@gmail.com). Eduardo Morales Sánchez investigador del Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada del IPN unidad Querétaro, México.,:

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Dado que es posible medir la actividad eléctrica del movimiento de los ojos o de los músculos implicados en el parpadeo o al cerrar los ojos de una forma relativamente sencilla, resulta atractivo desarrollar sistemas de comunicación para personas que no tienen un control absoluto de su cuerpo, ni del habla pero que aún tengan un gran control sobre sus ojos y los músculos implicados. Por ello se pensó en el desarrollo de un sistema de comunicación veloz que pueda ser controlado con el cierre y la apertura de los ojos.

II. Adquisición de Señal EOG

La adquisición de señal bioeléctrica ocular se realizó con un circuito integrado (CI) ADS1299 desarrollado por Texas Instruments para mediciones bioeléctricas, dicho circuito adquiere la señal en tiempo real de 8 canales en modo diferencial a una velocidad de muestre desde 250 muestras por segundo (MpS), hasta 16 mil MpS, dichas señales son amplificadas, convertidas en señales digitales y entregadas por protocolo SPI. (Figura 2)

A. Comunicación ADS1299 a microcontrolador

Dado que la frecuencia entre la que oscila la señal eléctrica ocular es de 0.3 a 38 Hz [4], el CI ADS1299 puede trabajar a su velocidad más baja (250 MpS), se utilizó un microcontrolador PIC 18F4550 esto debido a que cuenta con un módulo de SPI, la frecuencia de trabajo es de hasta 48 MHz y cuenta con un módulo de USB 2.0, lo cual es suficiente para la aplicación.

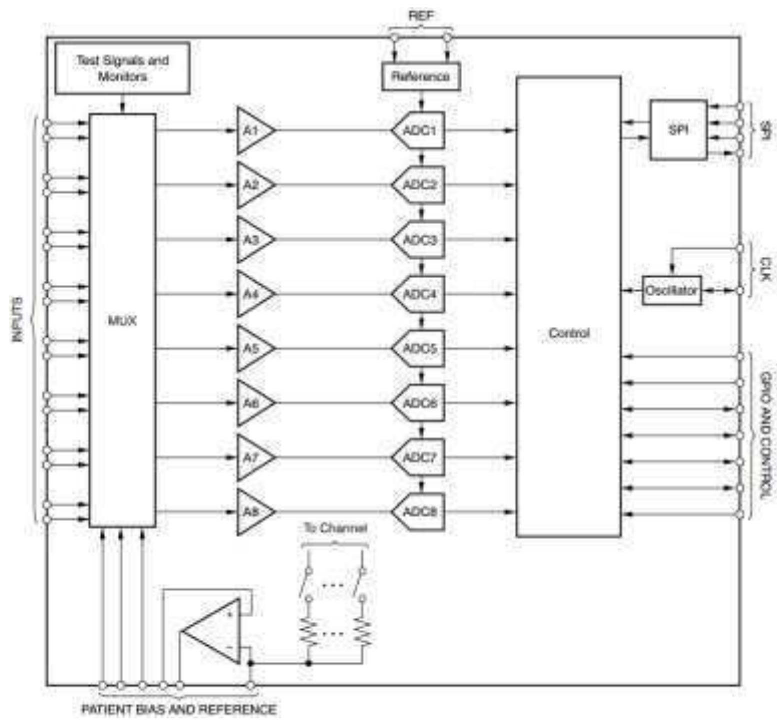


Figura 2 Diagrama de bloques del circuito integrado ADS1299

El PIC es el encargado de configurar el CI ADS1299, al cual se le debe especificar la ganancia de amplificación debido a que esta esta es programable, se debe ajustar el CI para que lea en modo diferencial, entre otros parámetros. Una vez que el CI se configura se adquiere la señal de 24 bits de los 8 diferentes canales. [5]

B. Comunicación microcontrolador a PC

Para realizar la comunicación entre el PIC y la PC se utilizó el software Labview y la capacidad del PIC de simular un puerto serial por el módulo USB. La señal que entrega el CI ADS1299 es enviada directamente a la PC en forma hexadecimal, sin realizar operaciones o algún procesamiento a esta señal, con la finalidad de que el PIC no llegue a perder datos de la señal.

Una vez que la señal es adquirida por el software Labview cada medición de cada canal debe ser transformada ya que esta se encuentra en complementos a 2, para lo cual, primero se debe convertir el valor a 32 bits ya que no existen variables de 24 bits, para esto se lee el bit 23, si éste es un 1 los espacios del 24 al 32 se completan con 1, posteriormente a la variable se le resta 1 y se niega todo el valor y se multiplica por -1, en el caso que el bit 23 sea 0, el valor corresponde sin necesidad de realizar alguna operación. (Figura 3)

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

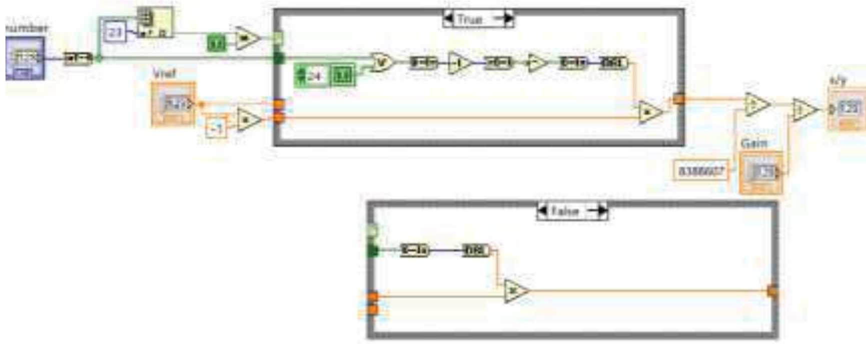


Figura 3 Complementos a 2 para 1 canal

C. Colocación de electrodos

Los electrodos se colocaron en forma vertical en ambos ojos (Figura 4) con la finalidad de registrar la actividad eléctrica de cada ojo y poder distinguir cuando se cierra un ojo u otro.

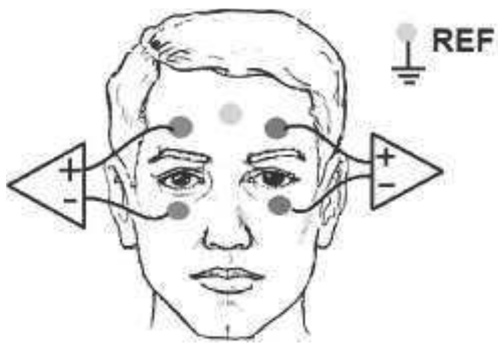


Figura 4 Posición de electrodos

Una vez que la señal es obtenida del PIC y transformada de complementos a dos se grafica en el tiempo para poder visualizarla (Figura 5), sin embargo, esta señal no es fácil de apreciar ni de interpretar debido a que es necesario acondicionarla, ya que la señal tiene ruido del ambiente que entra al cuerpo debido a que este actúa como una antena a cualquier señal, tal es el caso de la señal de la red eléctrica (60 Hz).

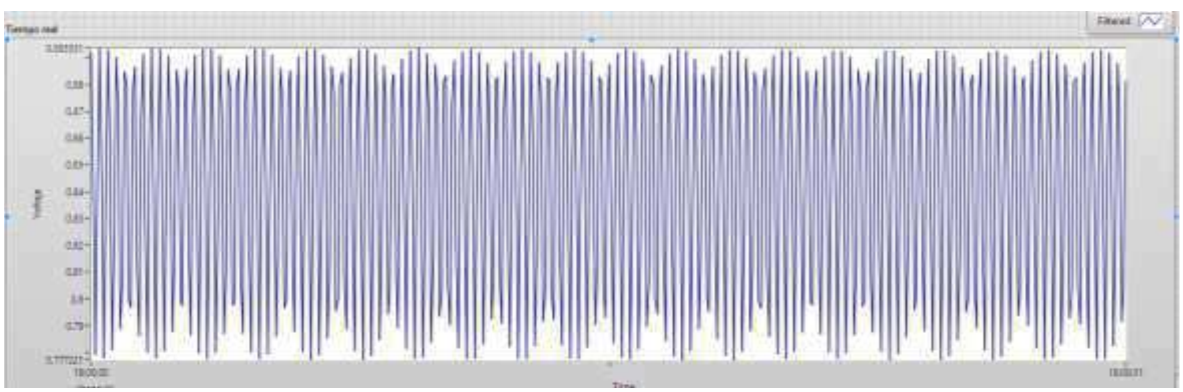


Figura 5 Señal en tiempo real

III. Acondicionamiento y Procesamiento de señal

A. Acondicionamiento de señal

Dado que el cuerpo humano funciona como una antena para toda señal en el ambiente es necesario aplicar filtros a la señal para acorar el rango de frecuencias que se encuentran en la señal, así como eliminar las frecuencias conocida como la de la red eléctrica (60 Hz).

Los rangos de frecuencia en los que oscila la señal eléctrica ocular se encuentra entre 0.3 a 38 Hz, sin embargo, para esta aplicación solamente es necesaria la señal correspondiente al parpadeo la cual se encuentra entre 1 y 3 Hz.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 14, páginas 140 - 146.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Para facilitar el procesamiento de la señal y dado que la señal del parpadeo se encuentra entre 1 y 3 Hz se aplicaron filtros a la señal en tiempo real, se aplicó un filtro mata-bandas a 60 Hz y un filtro pasa-banda de 1 a 20 Hz. (Figura 6)

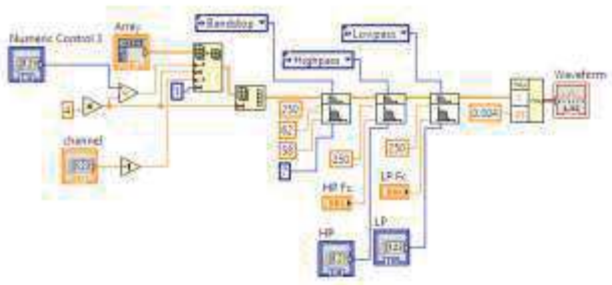


Figura 6 Filtros para acondicionar la señal

La señal que se obtiene después de realizar el acondicionamiento de la señal se puede ver en la Figura 7, en dicha figura se puede apreciar la actividad eléctrica del músculo ocular correspondiente al parpadeo, se pueden observar dos señales muy parecidas, esto debido a que se encuentran en ellas las señales correspondiente a la actividad eléctrica de ambos ojos.

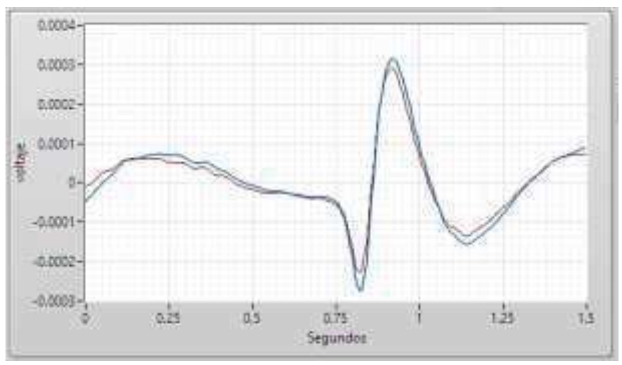


Figura 7 Señal acondicionada mostrando la actividad eléctrica del parpadeo

B. Procesamiento de señal

Se puede ver en la señal eléctrica del parpadeo que tiene una forma muy peculiar la cual es idéntica en cada parpadeo, al realizar una transformada rápida de Fourier se puede apreciar una frecuencia pico entre 1 y 3 Hz la cual corresponde a la señal del parpadeo (Figura 8).

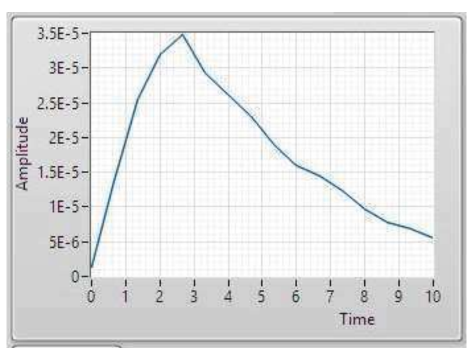


Figura 8 Transformada Rápida de Fourier de la Figura 7

Para poder realizar la Transformada Rápida de Fourier se tuvo que discretizar la señal en 1 segundo y medio por lo que cada señal de control de los parpadeos se realiza cada segundo y medio.

Dado que una Transformada Rápida de Fourier entrega una frecuencia pico entre 1 y 3 Hz cuando hay un parpadeo se utilizó esta técnica. A la señal de ambos ojos se aplicó tanto el acondicionamiento, como la transformada de Fourier.

Cuando se cierra y abre un solo ojo se puede observar en la gráfica de la señal de ambos ojos que ambas señales presentan un comportamiento similar, esto debido a que aunque sea pequeño en el ojo que no se cierra y abre existe un movimiento muscular el cual es detectado por el electrooculograma, cuando se analiza la señal, se puede apreciar que hay una señal que es

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

más pequeña y otra que es más grande, lo cual se puede ver reflejado en el espectro de frecuencia (Figura 9), esto nos indica que se realizó un parpadeo con un ojo.

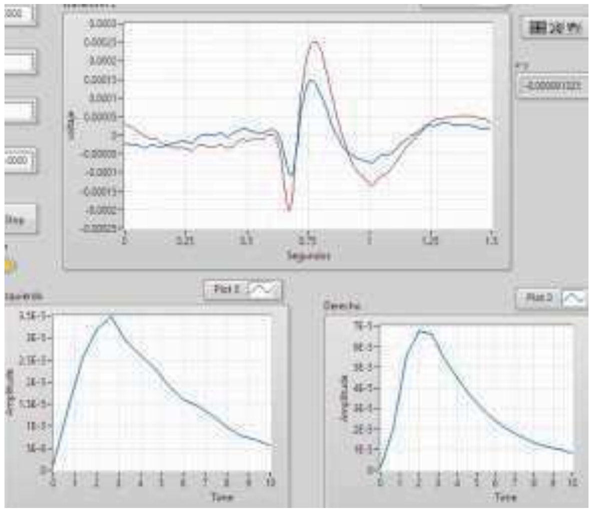


Figura 9 Parpadeo del ojo derecho

Dado que el espectro de frecuencia entrega una amplitud más grande en el espectro que corresponde al ojo con el que se parpadeo se decidió tomar esto como característica clave para poder clasificar entre el ojo derecho o el izquierdo.

Para realizar la clasificación, primeramente se analiza en el espectro de frecuencia de ambas señales, cual la frecuencia que tiene una amplitud mayor, si la frecuencia con mayor amplitud es entre 1.5 y 3 Hz en ambos espectros de frecuencia (ojos derecho y ojo izquierdo), el sistema supone que el usuario parpadeó o cerró y abrió un ojo, en caso contrario el sistema no actúa, cuando se detecta parpadeo o cierre y apertura del ojo, se comparan los valores de amplitud de la frecuencia con amplitud máxima, si la diferencia entre los valores es menor al 5% respecto al valor máximo el sistema considera que es un parpadeo normal, con ambos ojos y no se realiza acción de control, en caso que la diferencia sea mayor al 5% el sistema supone que un ojo se cerró y abrió y toma el que tuvo un valor máximo como verdadero.

En la Figura 9 el valor máximo de amplitud se encuentra en 2.5 Hz en el ojo izquierdo y en 2 Hz en el ojo derecho, por lo que el sistema considera que hubo un parpadeo, al comparar los valores picos, se encuentra que el ojo derecho tiene una amplitud mayor por lo que el sistema toma como cierre y apertura del ojo derecho.

IV. INTERFAZ PARA COMUNICACIÓN

Una vez que se logró clasificar las señales y que se comprobó que el sistema detecta y clasifica correctamente las señales, se desarrolló una interfaz para comunicación Hombre-Máquina, la interfaz se puede apreciar en la Figura 10, en la cual se tiene una barra central con números del 1 al 5, en esta el usuario puede rotar los números cerrando y abriendo los ojos para hacer que el número que quiere quede dentro del cuadro verde, estos números están relacionados con una frase las cuales se muestran en la parte inferior, una vez que el número que el usuario desea seleccionar se encuentra en el cuadro verde, la forma de seleccionar la frase se realiza con un botón el cual se puede colocar en cualquier sitio dependiendo de la movilidad del usuario; una vez que usuario selecciona un número, la frase relacionada a ese número se reproduce por las bocinas de la computadora y el sistema sigue trabajando de forma normal a la espera de otra señal de control.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.



Figura 10 Interfaz Hombre-Máquina

El sistema se probó con un usuario al que se le colocaron los electrodos como se muestra en la Figura 4 y se probó, se le pidió que seleccionara diferentes frases.



Figura 11 Colocación de los electrodos

Las pruebas realizadas mostraron un desempeño aceptable y un sistema muy sencillo de utilizar, sin embargo, se encontró que cuando se realizan movimientos rápidos del cuerpo o rotación de los ojos el sistema no logra identificar si se trata de un parpadeo o no.

V. Conclusión

Los resultado obtenidos, muestran un comportamiento aceptable ya que el sistema trabaja correctamente, con una velocidad de respuesta buena y suficiente para la aplicación, la solución al problema propuesta en este artículo, resulta muy interesante para el uso en personas con limitaciones para moverse y hablar, tal es el caso de personas tetrapléjicas que tienen un control muy limitado de su cuerpo.

Aunque el sistema es muy fácil de usar y una solución interesante, no resulta muy práctico al utilizar los electrodos, esto debido a que se pueden despegar después de un cierto tiempo y no se recomiendan para usarlos más de una vez, sin embargo se podría evaluar el uso de electrodos secos colocados en unas gafas, lo cual haría al sistema más práctico de usar, así como más cómodo.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

REFERENCIAS

[1] J. E. S. Gómez, El humano social anti-natural, España: Plaza y valdes, 2012.

[2] INEGI, «Discapacidad en México,» 2010.

[3] J. J. Kanski, Oftalmología Clínica, España: Elsevier, 2009.

[4] S. Kumar, A. Dash y M. Kumar M., «Design and Development of Low-Cost EOG Acquisition Circuit for HMI Application,» *2nd International Conference on Signal Processing and Integrated Networks* , pp. pp. 192 - 197, 2015.

[5] Texas Instruments, *Low-Noise, 8-Channel, 24-Bit Analog Front-End for Biopotential Measurements,* Texas, 2012.

[6] J. Catalá M. y M. Castany A., Pruebas electrofisiológicas, manual de instrucciónes, Barcelona: Thea Innovación, 2006.

[7] Á. L. Chávez R. y L. Corrales, «Diseño y construcción de un sistema de adquisición de datos para el análisis de los electrooculogramas,» *XIX Jornadas en ingeniería eléctrica y electrónica,* vol. 19, pp. pp. 74 - 79, 2005.

[8] C. Xingpan y Y. Pengpeng, «The front-end desing of portable EEG acquisition system based on the ADS1299,» *International Journal of Science and Research,* vol. 5, nº 3, pp. pp. 1513 - 1515, 2014.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 15, páginas 147 - 156.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Diseño de un robot móvil para desplazamiento en terreno agreste*

J. Alejandro Aguirre Anaya, O. Octavio Gutiérrez Frías

***Resumen*— El presente trabajo aborda el diseño de un robot móvil de tipo ruedas caminantes con sistema de suspensión pasiva en forma de paralelogramo con la capacidad de transitar sobre terreno poco estructurado, describiendo las características de la estructura mecánica, la instrumentación electrónica y la estrategia de control implementada en el sistema.**

I. Introducción

Un robot móvil requiere mecanismos que le permitan desplazarse sobre su entorno para realizar una determinada tarea o misión. En particular para solucionar el problema de movilidad sobre terreno agreste o poco estructurado, se propone el uso de patas ya que ofrecen la mejor maniobrabilidad con respecto a otro tipo de locomoción, sin embargo en terrenos planos son poco eficaces y requieren de un control sofisticado [1], por otra parte, se tienen los robots que se desplazan por medio de ruedas los cuales son eficientes en terreno plano pero no en terreno irregular, partiendo de esto, se tiene la alternativa de combinar la adaptabilidad de las patas con la eficiencia de las ruedas, y éstos se denominan robots móviles de tipo ruedas caminantes [1], existen dos tipos de robots de acuerdo a la complejidad del sistema de suspensión que utilicen: Los de suspensión activa y los de suspensión pasiva [2].

En el caso de los robots móviles de suspensión activa, se utilizan actuadores para modificar su centro de gravedad con la finalidad de mejorar su capacidad de superar obstáculos y su capacidad de adaptarse al terreno. En [3] se describe el robot SCARAB el cual emplea una articulación activa en los laterales para reconfigurar el sistema de suspensión. En [4] se presenta el robot denominado Octopus desarrollado por el Instituto Federal de Tecnología Suizo, el cual está formado por mecanismos en forma de paralelogramo con motores como actuadores, los cuales modifican el centro de gravedad de acuerdo al terreno donde transita, otros trabajos interesantes donde se aborda este tema se pueden encontrar en [5-7].

Por otra parte, los móviles de suspensión pasiva no tienen actuadores para modificar su centro de gravedad, el sistema de suspensión cambia dependiendo de la forma del terreno por la acción del mecanismo debido al peso, por tal motivo, requieren de pocos sensores y actuadores permitiendo que se tenga un consumo energético menor. Por ejemplo en [8] se describe el robot móvil Sojourner, que utiliza una suspensión llamada Rocker Bogie en los laterales con un sistema de tracción de seis ruedas independientes y cuatro motores para la dirección. En [9] se muestra un dispositivo que cuenta con 3 sistemas de suspensión independiente que tienen dos ruedas con tracción en cada uno de ellos denominado RCL-E. Además, en [10] los autores presentan el robot SOLERO el cual tiene una rueda en la parte frontal, una rueda en la parte posterior y dos sistemas de suspensión en cada lado cuya arquitectura es en forma de paralelogramo, otros ejemplos pueden encontrarse en [11-13].

El presente trabajo aborda el desarrollo de un robot móvil de tipo ruedas caminantes con sistema de suspensión pasiva, el cual será controlado de forma remota, utilizando un control de velocidad tipo control Proporcional Integral (PI) para cada articulación de las ruedas de tracción y un control Proporcional Integral Derivativo (PID) para mantener la dirección en una posición determinada, los cuales se han implementado de manera embebida en la tarjeta STM32F4 Discovery .

El trabajo está organizado de la siguiente manera: La sección II aborda las principales consideraciones del diseño del robot móvil. En la sección III se explica la instrumentación electrónica del robot. En la sección IV se explica la estrategia de control implementada y los resultados obtenidos. Finalmente, la última sección presenta las conclusiones del trabajo.

II. Diseño Mecánico del robot móvil

A. Metodología de diseño.

Para el diseño del robot móvil se siguen algunas etapas de la metodología propuesta en [14], las cuales se describen a continuación:

1. **Entradas de proceso:** Se requiere de un robot móvil con la capacidad de transitar sobre terreno poco estructurado que cubra con las necesidades establecidas en la tabla 1.

*Este trabajo es financiado por la Secretaria de Investigación y Posgrado del Instituto Politécnico Nacional (SIP-IPN), con número de registro 20164795.

J. Alejandro Aguirre está en la Sección de Estudios de Posgrado e Investigación, Unidad Profesional Interdisciplinaria en Ingeniería y Tecnologías Avanzadas, Instituto Politécnico Nacional, Ciudad de México, México, alejandroaguirre180983@hotmail.com.

Octavio Gutiérrez está en la Sección de Estudios de Posgrado e Investigación, Unidad Profesional Interdisciplinaria en Ingeniería y Tecnologías Avanzadas, Instituto Politécnico Nacional, Ciudad de México, México, ogutierrezf@ipn.mx

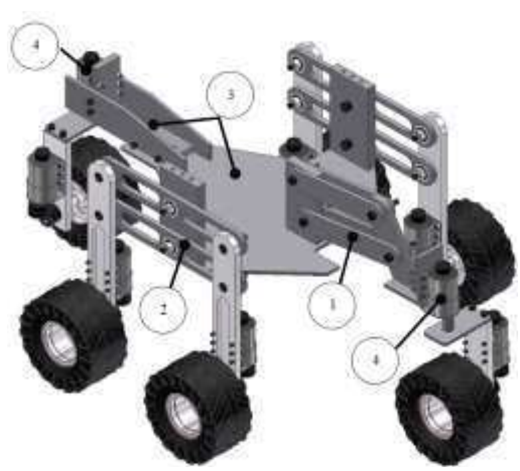
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Tabla 1. Entradas del proceso

Desplazamiento sobre terreno	El robot móvil debe ser capaz de transitar sobre diferentes tipos de terreno como concreto, grava, tierra y arena compacta.
Capacidad para superar obstáculos	Obstáculos de 120mm de altura.
Tipo de locomoción y suspensión del robot móvil	Se requiere un sistema adaptable y eficaz con un consumo mínimo de energía y facilidad en la implementación del control.
Selección de actuadores eléctricos	Tipo de motores a utilizar en la tracción y dirección
Restricciones de peso y dimensiones	Masa y tamaño del robot que permitan el movimiento así como la capacidad para subir y bajar pendientes de 35º y de transitar por terreno transversal con pendiente de 30º
Estrategia de control y tipo de operación	Se refiere a la manera en la cual será controlado el móvil.

2. **Estudios de requerimientos.** Esta etapa se realiza tomando como base las entradas del proceso, los resultados obtenidos se resumen a continuación:
    - **Desplazamiento sobre terreno.** Se seleccionaron llantas de 120mm de diámetro y 60mm de ancho fabricadas de caucho cuya superficie proporciona una tracción adecuada para diferentes tipos de terrenos, además de absorber vibraciones ocasionadas por las irregularidades del terreno.
    - **Capacidad para superar obstáculos.** Se seleccionó una configuración de suspensión pasiva en forma de paralelogramo en la parte frontal y en los laterales, la cual ha mostrado tener la mejor capacidad para superar obstáculos y un ejemplo de su aplicación es el robot SOLERO [10].
    - **Tipo de locomoción y suspensión del robot móvil.** Con la finalidad de combinar la adaptabilidad de las patas con la eficiencia de las ruedas, se seleccionó el sistema de locomoción denominado ruedas caminantes y con el fin de tener un consumo mínimo de energía y facilitar el control se selecciona una suspensión de tipo pasiva.
    - **Selección de actuadores eléctricos.** Para este caso se eligieron motores con caja de reducción de 131.25:1 de corriente directa alimentados a 12 V para la tracción y la dirección.
    - **Restricciones de peso y dimensiones.** Debido a que se requiere una masa menor a 15 Kg, se seleccionó el aluminio como material para la construcción ya que tiene propiedades mecánicas que facilitan el mecanizado, alta resistencia a la corrosión y densidad baja. Por otra parte, las dimensiones del móvil no deben superar 750mm de largo, 510 mm de ancho y 320mm de alto, con la finalidad de tener la capacidad adecuada para superar obstáculos y evitar volcaduras al transitar.
    - **Estrategia de control y tipo de operación.** En este caso la estrategia de control implementada en el robot móvil es por medio de juntas independientes, en la cual cada eje es controlado como un sistema de una entrada y una salida.
3. **Diseño conceptual.** En esta etapa se desarrolla la distribución de los principales componentes del robot móvil los cuales son: El mecanismo frontal (1), mecanismos laterales (2) cuerpo (3) y mecanismos de dirección (4) como se muestra en la figura 1.

Figura 1. Componentes mecánicos del robot móvil.



4. **Diseño detallado.** A lo largo del trabajo se realiza la descripción de las principales partes que componen al robot móvil.
5. **Salida de proceso.** A partir del diseño de la figura 1 se pueden crear los planos para la manufactura del robot móvil.

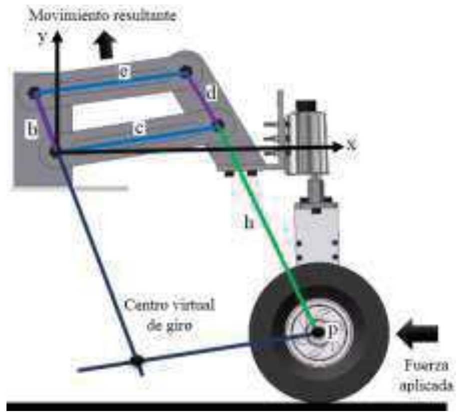
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Por lo anterior, a partir de la metodología descrita se puede desarrollar un robot móvil para desplazamiento en terreno agreste y a continuación en cada una de las secciones se describen los aspectos más relevantes para llevarlo a cabo.

B. Mecanismo frontal

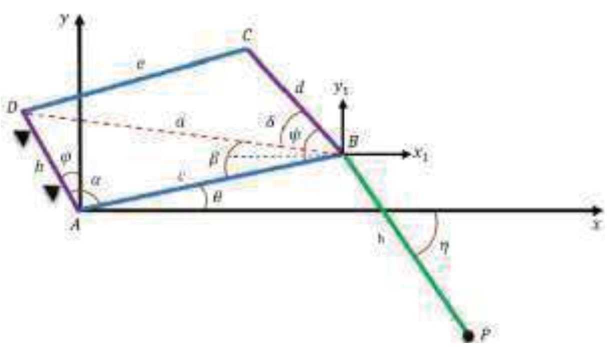
El mecanismo frontal cumple con dos tareas: Garantizar el contacto de la rueda con el piso en todo momento y elevarla cuando el móvil se encuentra con un obstáculo para poder superarlo [15]. Para cumplir con esas tareas, se utiliza un mecanismo de cinco barras mediante el cual se genera un pivote o centro virtual de giro situado debajo del eje de la rueda como se muestra en la figura 2.

Figura 2. Configuración del mecanismo frontal.



Por otra parte, para obtener la configuración que permita superar obstáculos de 120 mm, se realiza un análisis geométrico para conocer la posición del punto P localizado en el eje de la rueda delantera de acuerdo a la figura 3.

Figura 3. Análisis geométrico del mecanismo frontal



De la figura 3, se obtienen las siguientes ecuaciones:

$$\alpha(\theta) = \varphi + \frac{\pi}{2} - \theta \quad (1)$$

$$\psi(\theta) = \beta + \delta \quad (2)$$

$$a = \sqrt{b^2 + c^2 - 2bc\text{Cos}(\alpha(\theta))} \quad (3)$$

$$\beta = \arccos\left(\frac{a^2 + c^2 - b^2}{2ac}\right) \quad (4)$$

$$\delta = \arccos\left(\frac{a^2 + d^2 - e^2}{2ad}\right) \quad (5)$$

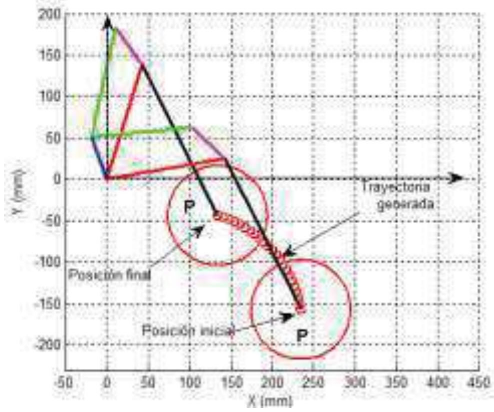
$$\eta(\theta) = \psi(\theta) - \theta \quad (6)$$

$$P(\theta) = \begin{cases} c\cos(\theta) + h\cos(\eta(\theta)) \\ c\sin(\theta) - h\sin(\eta(\theta)) \end{cases} \quad (7)$$

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 15, páginas 147 - 156.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Utilizando de (1) a (7), se propone el desplazamiento y la trayectoria generada al elevar la rueda de manera suave, para ello se realizó una simulación numérica que se muestra en la figura4 con lo cual se determinaron los valores de los eslabones que forman el mecanismo frontal cuando chocan con un obstáculo de 120 mm los cuales son: que son $\mathrm{b} = 55\mathrm{mm}$ $\mathrm{c} = 145\ \mathrm{mm}$, $\mathrm{d} = 55mm$, $\mathrm{e} = 135\ \mathrm{mm}$ y $\mathrm{h} = 204\ \mathrm{mm}$.

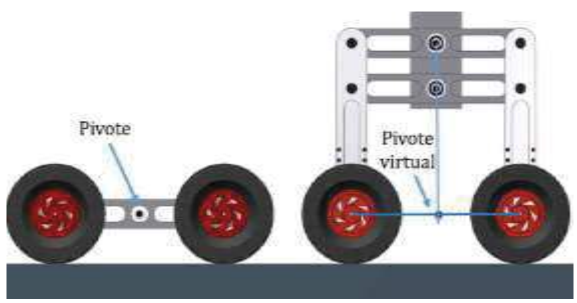
Figura 4. Trayectoria generada del mecanismo frontal



C Mecanismo lateral

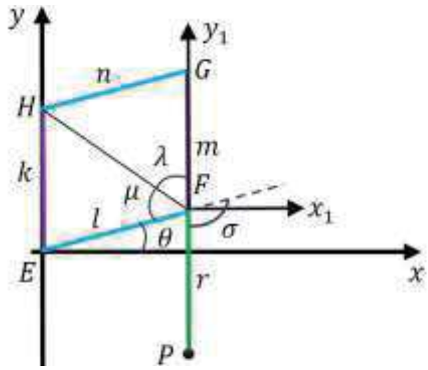
La suspensión lateral es el principal elemento que integra al robot móvil ya que proporciona estabilidad durante el movimiento incluso al transitar sobre terreno poco estructurado. Para asegurar una adecuada capacidad de adaptación al terreno, es necesario mantener el pivote de giro lo más bajo posible y al mismo tiempo mantenerlo libre de obstáculos [16]. Se propone una configuración en forma de paralelogramo con lo que se crea un pivote virtual que cumple con las características anteriormente mencionadas, la figura 5 muestra la suspensión propuesta.

Figura 5. Sistema de suspensión en forma de paralelogramo



De acuerdo a la figura 6, se obtiene la posición del punto P localizado en el eje de la rueda derecha delantera del mecanismo lateral, para simplificar el cálculo se toma la mitad del sistema a partir de los puntos de apoyo E y H que son fijos y se obtiene (8).

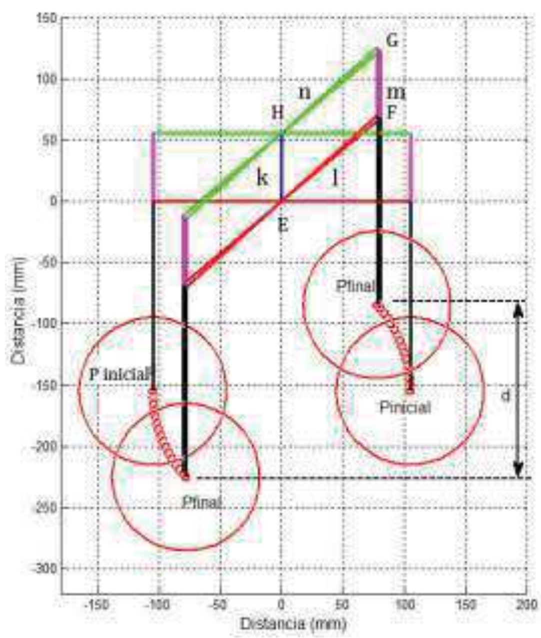
Figura 6. Análisis geométrico del mecanismo lateral



$$P(\theta) = \begin{cases} l\cos(\theta) \\ l\sin(\theta) - r \end{cases} \quad (8)$$

Mediante simulación numérica se obtienen los valores para los eslabones k, l, m, n y r que son: 56mm, 105mm, 56mm 105mm y 154mm respectivamente En la figura 7, se muestra la posición inicial y final del mecanismo lateral cuando se presenta un obstáculo $d = 120mm$

Figura 7 Análisis geométrico del mecanismo lateral



D Margen de estabilidad

La estabilidad del robot móvil en forma estacionaria o a velocidad constante se puede definir con el margen de estabilidad gravitacional, el cual es la distancia mínima del centro de gravedad proyectado sobre el plano de tierra al borde definido por el contacto de las dos ruedas [17]. La pendiente máxima que el robot móvil puede escalar manteniendo la estabilidad se llama estabilidad longitudinal y se obtiene mediante (9). Si transita a lo largo de una pendiente se denomina estabilidad transversal y se obtiene mediante (10).

$$\varphi = \min\left\{\arctan\left(\frac{L_1}{Hl}\right), \arctan\left(\frac{L_2}{Hl}\right)\right\} \quad (9)$$

$$\alpha = \min\left\{\arctan\left(\frac{d_1}{Ht}\right), \arctan\left(\frac{d_2}{Ht}\right)\right\} \quad (10)$$

Tomando en cuenta las características de los mecanismos frontal y lateral del robot móvil y con el fin de cumplir con las dimensiones y una distribución uniforme del peso, se tiene que $L_1 = 303mm$, $L_2 = 330mm$, $d_1 = d_2 = 202mm$ y $Hl = Ht = 188mm$, obteniendo lo siguiente:

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

$$\varphi = \min\left\{\arctan\left(\frac{303\text{mm}}{188\text{mm}}\right), \arctan\left(\frac{330\text{mm}}{188\text{mm}}\right)\right\} = 58.1° \quad (11)$$

$$\alpha = \min\left\{\arctan\left(\frac{202\text{mm}}{188\text{mm}}\right), \arctan\left(\frac{202\text{mm}}{188\text{mm}}\right)\right\} = 47° \quad (12)$$

De las expresiones (11) y (12) se observa que para la estabilidad longitudinal y transversal, se tiene un factor de seguridad de 1.66 y 1.56 respectivamente, por lo que se cumplen con los requerimientos de diseño y se evitan volcaduras. En la figura 8 se muestra el margen de estabilidad longitudinal y en la figura 9 el margen de estabilidad transversal

Figura 8. Margen de estabilidad longitudinal

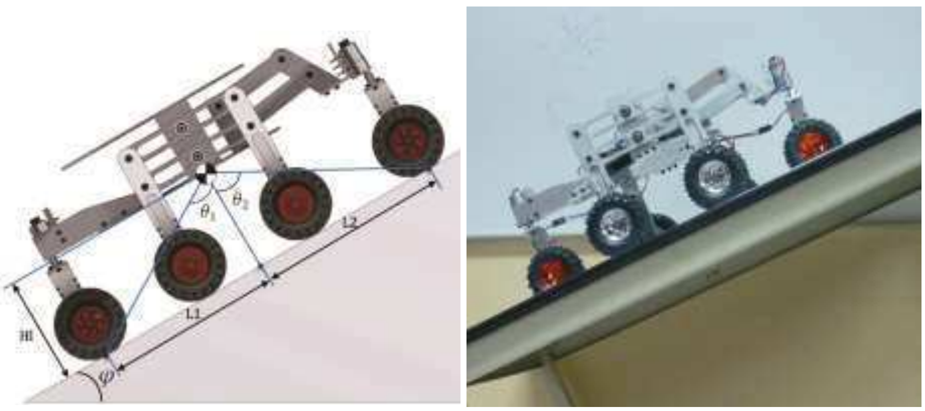
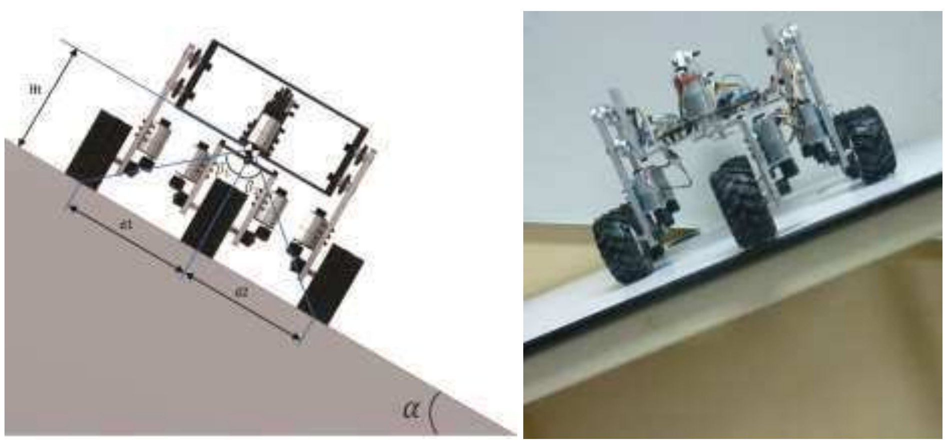


Figura 9. Margen de estabilidad transversal.



III. SISTEMA ELECTRÓNICO.

El sistema electrónico se divide en dos, la estación base y la estación móvil. En la primera se tiene una computadora con una interfaz y un módulo de transmisión inalámbrica mediante, los cuales el usuario manda las instrucciones al robot móvil. En la estación móvil, se tienen dos tarjetas de control STM32F4, una configurada como maestro y otra como esclavo.

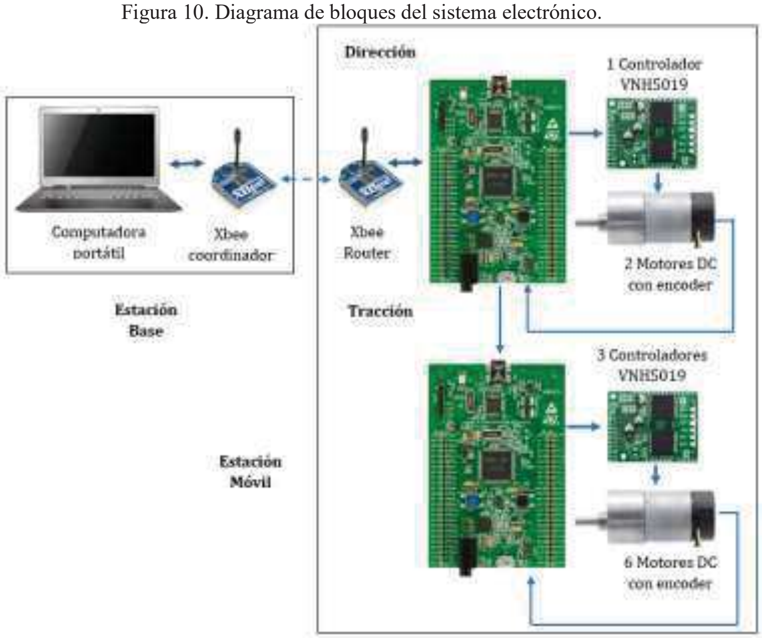
La tarjeta configurada como maestro se encarga de recibir la información del transmisor inalámbrico, de la lectura de los codificadores ópticos de los motores de tracción, de enviar a la tarjeta configurada como esclavo las señales de posición, además de que tiene programados los seis controladores de velocidad para los motores de tracción.

La tarjeta configurada como esclavo lee los codificadores ópticos de los motores de dirección y tiene programado los controladores PID de posición. La figura 10 muestra el diagrama esquemático

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Los motores utilizados para la tracción y la dirección tienen una caja reductora de relación 131.25 a 1 con un torque de 18 Kg*cm y velocidad de 80 rpm, el voltaje de alimentación es de 12 V y la corriente máxima con torque es de 5 A. Cuentan con codificadores ópticos de 8400 cuentas por revolución y utilizan los drivers VNH5019 Dual como etapa de potencia ya que pueden conectarse motores de 5.5 V a 24 V con una corriente máxima de 12 A por canal.

Figura 10. Diagrama de bloques del sistema electrónico.



IV. ESTRATEGIA DE CONTROL.

Se propone un controlador de velocidad para cada motor de tracción y un control de posición para los motores de dirección.

En la figura 11 se muestra el diagrama de bloques implementado en la tarjeta STM32F4 Discovery con la ayuda de las librerías de Waijung de la compañía Aimaigin para uno de uno de los motores. Se describe a continuación los elementos que lo componen:

1. Bloque de configuración de la tarjeta. Sirve para indicar el modelo de tarjeta a utilizar, el tipo de programador y depurador.
2. Bloque de lectura del codificador óptico. Este bloque lee los pulsos generados por el movimiento del motor.
3. Bloque de acondicionamiento. Se utiliza para convertir el número de cuentas generado por el codificador óptico en revoluciones por minuto para el control de velocidad o en grados para el control de posición.
4. Bloque de control. Si el motor es de tracción contiene el controlador de velocidad Proporcional Integral PI como se muestra en la figura 12(a), de otra manera contiene el control Proporcional Integral Derivativo PID mostrado en la figura 12(b) para la dirección.
5. Bloque PWM. Realiza la modulación por ancho de pulso que permite variar la velocidad del motor.
6. Bloque de salidas digitales. Se encarga de mandar la dirección de giro del motor, en sentido horario o antihorario.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Figura 11. Diagrama a bloques del control implementado por motor.

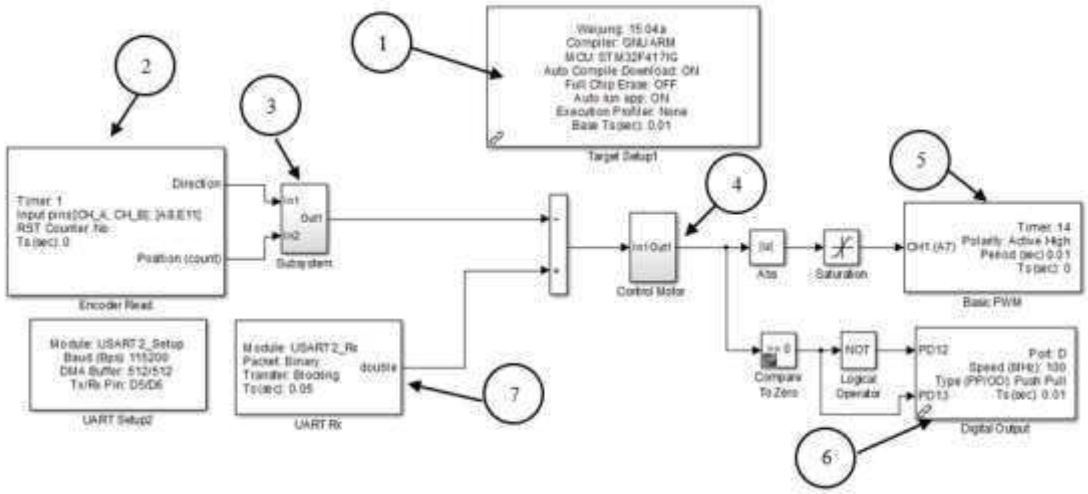
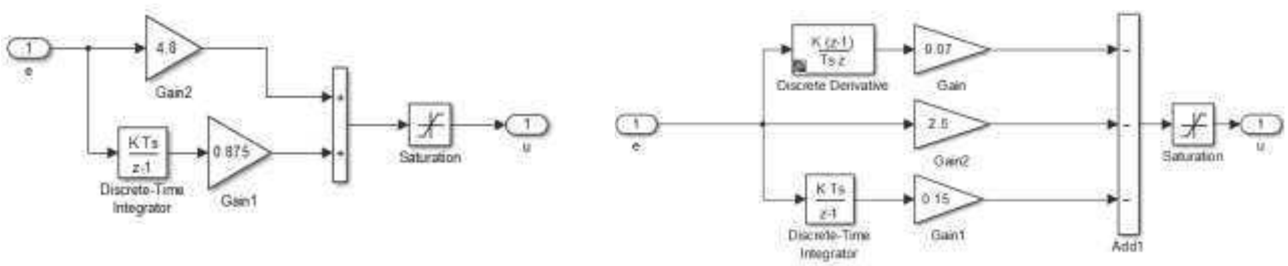


Figura 12. Controlador Proporcional Integral PI (izquierda), controlador Proporcional Integral Derivativo (derecha)

a) *b)*



En la figura 12 se muestra el diagrama de bloques del controlador Proporcional Integral y el controlador proporcional Integral Derivativo cuyas expresiones matemáticas son (13) y (14).

$$\mathrm{U}_{vel}(\mathrm{Z}) = [\mathrm{K}_{\mathrm{p1}} + \mathrm{K}_{\mathrm{I1}}(1 - \mathrm{Z}^{-1})]\mathrm{E}(\mathrm{Z}) \qquad (13)$$

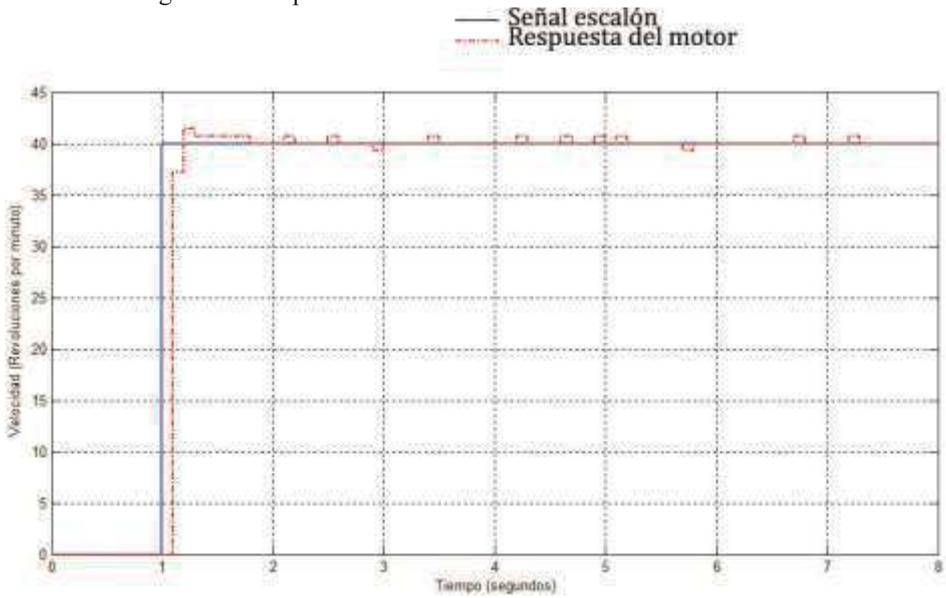
$$U_{pos}(Z) = [K_{p2} + K_{D2}\left(\frac{1}{1 - Z^{-1}}\right) + K_{I2}(1 - Z^{-1})]E(Z) \qquad (14)$$

Para verificar el funcionamiento se realizaron dos experimentos uno para tracción y otro para la dirección con los controladores propuestos implementados en la tarjeta embebida donde los parámetros K_{p1} y K_{I1} para el controlador de velocidad son 4.8 y 0.875 respectivamente y para el control de dirección se tiene que $K_{p2} = 2.5$, $K_{D2} = 0.07$ y $K_{I2} = 0.15$

En la figura 13 se observa la respuesta de uno de los motores de tracción ante una señal escalón cuya amplitud es de 40 RPM, en la gráfica se observa que la respuesta converge a la referencia en un tiempo adecuado y con valor de error de 1RPM.

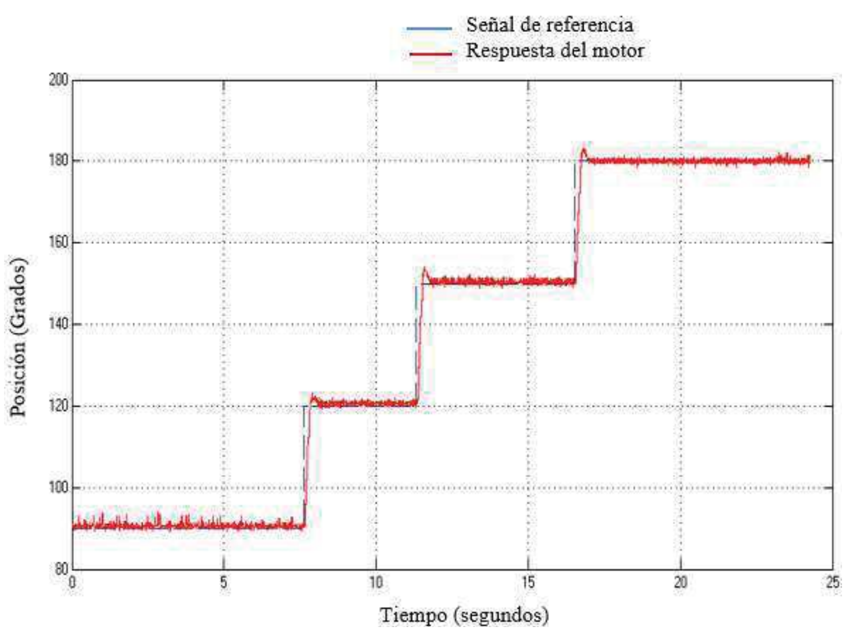
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Figura 13. Respuesta del motor de tracción a una señal escalón.



En la figura 14, se observa el resultado obtenido al aplicar el controlador de posición en uno de los motores de dirección, en este caso, se realizó la prueba con una señal en forma de escalera en la cual se varió el ángulo de 90 a 180 grados y se observó que el controlador seguía la señal de error promedio de 1.5 grados.

Figura 14. Respuesta del motor de dirección.



V. Conclusiones.

La metodología de diseño implementada puede ser aplicada para el diseño de robots móviles dedicados a la exploración de terreno poco estructurado facilitando tareas de exploración. Por otra parte con el análisis geométrico realizado en el mecanismo lateral y frontal se pueden encontrar numéricamente el valor de cada uno de los eslabones que los constituyen con la finalidad de que cumplan con los criterios establecidos de superación de obstáculos y margen de estabilidad

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

transversal y longitudinal. Por último, los controladores implementados en la tarjeta de desarrollo STM32F4 son adecuados para el control de tracción y dirección del robot móvil.

REFERENCIAS

[1] R. Siegwart, I. Nourbakhsh, and D. Scaramuzza, *Introduction to autonomous mobile robots*. MIT Press, 2011 ch 2.

[2] R. Siegwart, P. Lamon, T. Estier, M. Lauria, and R. Piguet, "Innovative design for wheeled locomotion in rough terrain," *Robotics and Autonomous systems*, vol. 40, no 2, pp. 151-162, 2002.

[3] P. Bartlett, D. Wettergreen, and W. L. Whittaker, "Design of the Scarab Rover for Mobility and Drilling in the Lunar Cold Traps". In: International Symposium on Artificial Intelligence, Robotics and Automation in Space, 2008

[4] M. Lauria, Y. Piguet and R. Siegwart, "Octopus an autonomous wheeled climbing robot," Fifth International conference on climbing and walking robots 2002.

[5] I. Leppanen, S. Salmi, and A. Halme, "WorkPartner HUT Automation's new hybrid walking machine," In *1st International Conference on Climbing and Walking Robots, Brussels*. p. 391-394. 1998.

[6] M. Van Winnendael, G. Visenti R, Bertrand and R. Rieder, "Nanokhod microrover heading towards Mars," *European Space Agency-Publications-ESA SP*, vol. 440, p. 69-76, 1999.

[7] C. Granda, F. Ben and P. Bidaud," "Kinematic Analysis and stability optimization of a reconfigurable legged-wheeled mini rover," in *AeroSense 2002*. International Society for Optics and Photonics, pp. 295-302, 2002.

[8] R. Team, "Characterization of the Martian surface deposits by the Mars Pathfinder rover, Sojourner". *Science*, vol. 278, no 5344, pp. 1765-1768, 1997.

[9] V. Kucherenko, A. Bogatchev, and M. Van Winnendael, "Chassis concepts for the ExoMars rover," In *The 8th ESA Workshop on Advanced Space Technologies for Robotics and Automation,* 2004.

[10] S. Michaud, A. Schneider, R. Bertrand, P. Lamon, R. Siegwart, M. Van Winnendael, and A. Schiele, "SOLERO: Solar powered exploration rover". In *None* (No. LSA-CONF-2002-004), 2002.

[11] R. Volpe, J. Balaram, T. Ohm and R. Ivlev, "Rocky 7: "A next generation Mars rover prototype", *Advanced Robotics*, vol. 11, no 4, pp. 341-358, 1996

[12] T. Thueer, P. Lamon, A. Krebs and R. Siegwart, "CRAB-Exploration rover with advanced obstacle negotiation capabilities," Swiss Federal Institute of Technology (ETHZ), Autonomous Systems Lab, 2006.

[13] T. Kubota, Y. Kuroda, Y. Kunii and I. Natakani, "MICRO PLANETARY ROVER" In *Proceedings of Fifth International Symposium on Artificial Intelligence. Robotics and Automation in Space (ESA SP-440)*, pp. 373-378, 1999.

[14] D. Apostolopoulos, *Systematic Configuration of Robotic Locomotion tech*. report CMU-RI-TR-96-30, Robotics Institute, Carnegie Mellon University.

[15] S. Dian, T. Liu, Y. Liang, M. Liang, & W. Zhen, "A novel shrimp rover-based mobile robot for monitoring tunnel power cables" En *2011 IEEE International Conference on Mechatronics and Automation*. IEEE, p. 887-892, 2011.

[16] T. Estier, R. Piguet, R. Eichhorn and R. Siegwart, R. "Shrimp, a rover architecture for long range martian mission," In *Proceedings of the Sixth ESA Workshop on Advanced Space Technologies for Robotics and Automation,* pp. 5-7, 2000.

[17] D. APOSTOLOPOULOS, Analytic configuration of wheeled robotic locomotion. 2001.

[18] K. Ogata, *Sistemas de control en tiempo discreto*. Pearson Educaciòn, 1996 pp 114-117.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 16, páginas 157 - 162.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Determinación analítica del balance estático de un robot bípedo *

Christian Alberto Matilde Domínguez, Eduardo Morales Sánchez

***Resumen*— Este trabajo describe una metodología para determinar el balance estático de un robot bípedo durante la ejecución de un ciclo de marcha asimétrico, de manera analítica. El robot es modelado como un árbol cinemático. La localización del robot se describe con respecto a un sistema de referencia adjunto al entorno, la posición del centro de masa del robot, así como la posición de su base de soporte se determina con respecto a este sistema.**

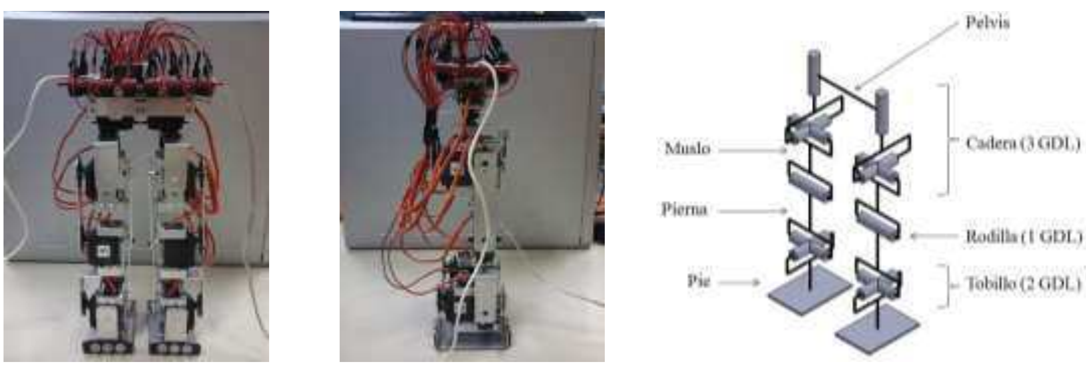
I. Introducción

CICATA Querétaro se planteó, como objetivo a largo plazo, el desarrollo de un robot humanoide de marcha dinámicamente balanceada, capaz de operar en ambientes que han sido creados para los humanos. En la primera etapa del proyecto se desarrolló un robot bípedo de marcha estáticamente balanceada [1]. En este trabajo se presenta la metodología que se utiliza para determinar de manera analítica el balance estático de este robot (fig. 1), durante la ejecución de un ciclo de marcha asimétrico, cuando su localización se describe con respecto a un sistema de coordenadas adjunto al entorno.

Para realizar la determinación del balance estático del robot es necesario:

1. Obtener el modelo geométrico directo del robot.
2. Determinar la posición del centro de masa del robot.
3. Determinar la localización del robot con respecto al entorno.

En la sección A se describe la metodología para obtener el modelo geométrico directo del robot. En la sección B se presenta la ecuación que define la posición del centro de masa del robot, con respecto a un sistema de coordenadas adjunto a uno de sus eslabones. En la sección C se mencionan las fases de la marcha bípeda. En la sección D se exponen las ecuaciones para describir la localización del robot con respecto al entorno. En la sección E se presentan las ecuaciones para evaluar si se satisface la condición de balance estático durante la marcha. Por último, se describe la metodología para determinar el balance estático del robot de forma analítica.



(a) (b) (c)

Figura 1. Robot bípedo de 12 grados de libertad [1]. (a) Vista frontal. (b) Vista lateral. (c) Eslabones y articulaciones.

II. Requerimientos

A. Modelo geométrico directo del robot

Se define como el conjunto de matrices de transformación homogénea que se utiliza para determinar la pose de los eslabones del robot con respecto al eslabón base, a partir de los vectores de articulación $\Theta = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6]^{\mathrm{T}}$ y $\Theta' = [\theta'_1, \theta'_2, \theta'_3, \theta'_4, \theta'_5, \theta'_6]^T$. La metodología para obtener este modelo consiste en:

1. Asignar sistemas de coordenadas a los eslabones del robot.
2. Determinar los parámetros que relacionan a los sistemas adjuntos.
3. Determinar las matrices de transformación individual de las extremidades.
4. Determinar la pose de cada eslabón con respecto al eslabón base.

*Esta investigación se realiza gracias al apoyo de CONACYT.
Christian Alberto Matilde Domínguez es alumno del Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada Unidad Querétaro, México (e-mail: chris_imec@hotmail.com).
Eduardo Morales Sánchez es profesor investigador del Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada Unidad Querétaro, México (e-mail: emoraless@ipn.mx).

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Se asignan sistemas de coordenadas a los eslabones del robot, como se muestra en la fig. 2, siguiendo la convención descrita por [2]. Los sistemas de coordenadas se asignan considerando al robot como un árbol cinemático [3] [4]. Después, se determinan los parámetros de Denavit-Hartenberg que describen la geometría de los eslabones del robot y la forma en que se conectan (tablas 1 y 2). Se evalúan estos parámetros en la ecuación siguiente [2]:

$$ {}^{i-1}_{\ \ i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1} d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & -s\alpha_{i-1} & c\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{1} $$

para obtener las matrices de transformación individual de las extremidades derecha ($i = 1,2,\dots,6$) e izquierda ($i = 1', 2', \dots, 6'$). Por último, se evalúa la ecuación siguiente ($i = 2,3,\dots,6,2',3',\dots,6'$):

$$ {}^{0}_{i}T = {}^{\ \ \ 0}_{i-1}T \ {}^{i-1}_{\ \ i}T, \tag{2} $$

para determinar las matrices de transformación homogénea restantes, que describen la pose de los eslabones 2 a 6 y 2' a 6' con respecto al eslabón 0. Las matrices que describen la pose de los eslabones 1 y 1', con respecto al eslabón 0, se obtienen a partir de la ecuación (1).

El modelo geométrico directo del robot se expresa, en forma compacta, como $MGD = \{{}^{0}_{i}T\}_{i=1,2,\dots,6,1',2',\dots,6'}$. La matriz ${}^{0}_{i}T$ se utiliza para determinar la posición de un punto P, relativa al sistema de coordenadas 0, cuando se conoce su posición con respecto a otro sistema de coordenadas ($i = 1,2,\dots,6,1',2',\dots,6'$):

$$ {}^{0}P = {}^{0}_{i}T \ {}^{i}P. \tag{3} $$

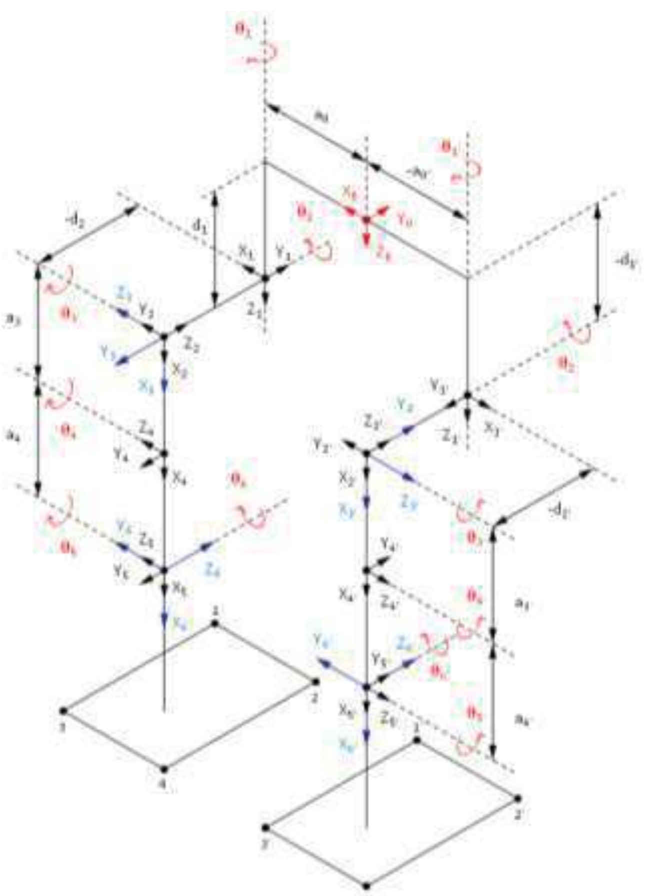


Figura 2. Asignación de sistemas de coordenadas a los eslabones del robot, cuando éste se encuentra en la postura de referencia definida por los vectores $\Theta = [0°, -90°, 0°, 0°, 0°, 0°]^T$ y $\Theta' = [180°, 90°, 0°, 0°, 0°, 0°]^T$.

Tabla 1. Parámetros de Denavit-Hartenberg de la extremidad derecha.

i	α_{i-1} [°]	a_{i-1} $[mm]$	d_i $[mm]$	θ_i [°]
1	0	a_0	d_1	θ_1
2	-90	0	$-d_2$	θ_2
3	-90	0	0	θ_3
4	0	a_3	0	θ_4
5	0	a_4	0	θ_5
6	90	0	0	θ_6

Tabla 2. Parámetros de Denavit-Hartenberg de la extremidad izquierda.

i'	$\alpha_{i'-1}$ [°]	$a_{i'-1}$ $[mm]$	$d_{i'}$ $[mm]$	$\theta_{i'}$ [°]
$1'$	0	$-a_{0'}$	$d_{1'}$	$\theta_{1'}$
$2'$	90	0	$-d_{2'}$	$\theta_{2'}$
$3'$	90	0	0	$\theta_{3'}$
$4'$	0	$a_{3'}$	0	$\theta_{4'}$
$5'$	0	$a_{4'}$	0	$\theta_{5'}$
$6'$	-90	0	0	$\theta_{6'}$

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 16, páginas 157 - 162.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

B. Centro de masa del robot

El vector de posición del centro de masa (CM) del robot, definido con respecto al sistema de coordenadas 0, está dado por la ecuación siguiente:

$$ {}^{0}CM = \frac{m_0\ {}^{0}CM_0 + \sum_i^{6'} m_i\ {}^{0}T_i\ {}^{i}CM_i}{m_0 + \sum_i^{6'}\ {}^{0}m_i}, \tag{4} $$

donde $i = 1,2,\dots,6,1',2',\dots,6'$. El vector ${}^{i}CM_i$ define la posición del centro de masa del eslabón i-ésimo con respecto al sistema de coordenadas que se le adjuntó.

C. Fases de la marcha bípeda

El ciclo de marcha de una extremidad se compone de 4 fases: primer doble soporte (fase 1), soporte simple (fase 2), segundo doble soporte (fase 3) y balanceo (fase 4). Al caminar, los ciclos de marcha de las extremidades se desarrollan de manera simultánea (fig. 3), de modo que existe una equivalencia entre las fases de ambos, por lo tanto, la marcha bípeda puede considerarse como una sucesión alternativa de las fases consecutivas de primer doble soporte y de soporte simple de las extremidades [5].

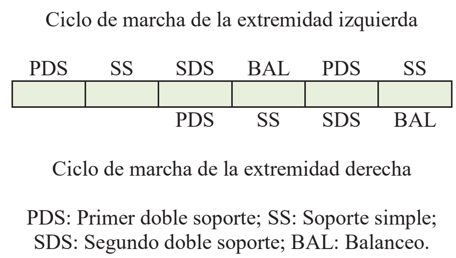


Figura 3. Equivalencia entre las fases de los ciclos de marcha de las extremidades derecha e izquierda.

Un ciclo de marcha puede ser asimétrico o simétrico. En el primer caso, el robot se comporta de forma única durante el desarrollo de cada fase, mientras que en el segundo, su comportamiento durante la ejecución de las fases 1 y 2 es equivalente al que exhibe al realizar las fases 3 y 4.

D. Localización del robot con respecto al entorno

Considérese el vector de posición ${}^{0}P$ de un punto P. El punto P puede estar definido con respecto al sistema de coordenadas 0 u otro, como se estableció en la ecuación (3). La posición del punto P con respecto a un sistema de referencia A adjunto al entorno, está dada por:

$$ {}^{A}P = {}^{A}_{6}T\ {}^{6}_{0}T\ {}^{0}P, \tag{5} $$

cuando los eslabones del robot se mueven con respecto al pie derecho (fases de primer doble soporte y de soporte simple de la extremidad derecha), y por:

$$ {}^{A}P = {}^{A}_{6'}T\ {}^{6'}_{0}T\ {}^{0}P, \tag{6} $$

cuando se mueven con respecto al pie izquierdo (fases de primer doble soporte y de soporte simple de la extremidad izquierda).

Las ecuaciones (5) y (6) se utilizan para determinar la posición, con respecto al sistema de referencia A, de cualquier punto definido con respecto a un sistema de coordenadas adjunto a uno de los eslabones del robot. De modo que estas ecuaciones permiten:

1. Determinar la pose de los eslabones del robot con respecto al entorno.
2. Determinar la trayectoria que el centro de masa del robot describe con respecto al entorno, cuando éste ejecuta un ciclo de marcha.
3. Determinar la base de soporte del robot con respecto al entorno, durante la ejecución de un ciclo de marcha.

E. Evaluación del balance estático durante la marcha

Las figs. 4 a 7 muestran la base de soporte del robot durante el desarrollo de un ciclo de marcha asimétrico de la extremidad izquierda. La proyección del centro de masa del robot, sobre el suelo, está definida por el punto P de coordenadas (x,y). La condición de balance estático se cumple cuando el punto P pertenece a la base de soporte del robot [5], es decir, cuando las coordenadas (x,y) satisfacen las 4 desigualdades siguientes, durante el desarrollo de las 4 fases del ciclo de marcha, respectivamente:

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 16, páginas 157 - 162.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

$$x_1 \leq x \leq x_{2'} \qquad Y_3 \leq y \leq Y_1,$$
$$x_{2'} < x < x_3 \qquad Y_3 \leq y \leq Y_2, \tag{7}$$
$$x_3 \leq x \leq x_{4'} \qquad Y_4 \leq y \leq Y_2;$$

$$x_{1'} \leq x \leq x_{4'} \qquad Y_5 \leq y \leq Y_2; \tag{8}$$

$$x_{1'} \leq x \leq x_2 \qquad Y_7 \leq y \leq Y_2,$$
$$x_2 < x < x_{3'} \qquad Y_8 \leq y \leq Y_2, \tag{9}$$
$$x_{3'} \leq x \leq x_4 \qquad Y_8 \leq y \leq Y_6;$$

$$x_1 \leq x \leq x_4 \qquad Y_8 \leq y \leq Y_9. \tag{10}$$

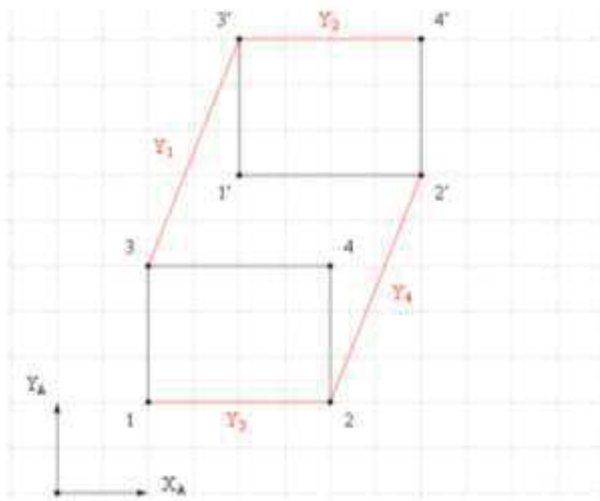


Figura 4. Base de soporte del robot durante la fase de primer doble soporte de la extremidad izquierda.

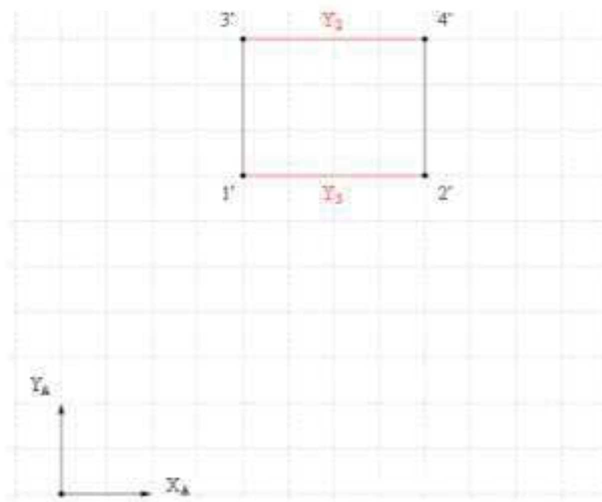


Figura 5. Base de soporte del robot durante la fase de soporte simple de la extremidad izquierda.

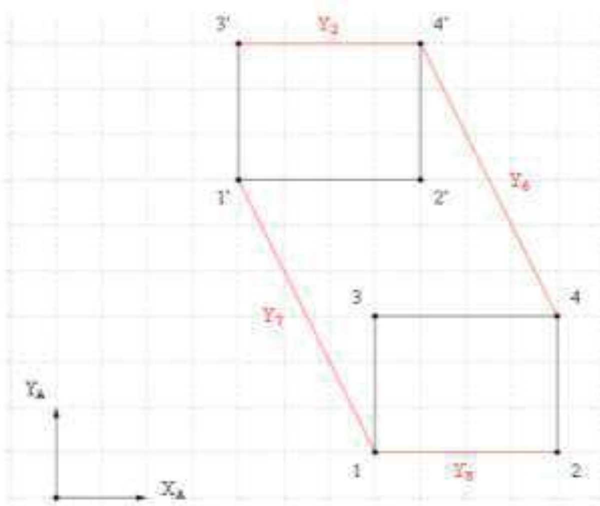


Figura 6. Base de soporte del robot durante la fase de segundo doble soporte de la extremidad izquierda.

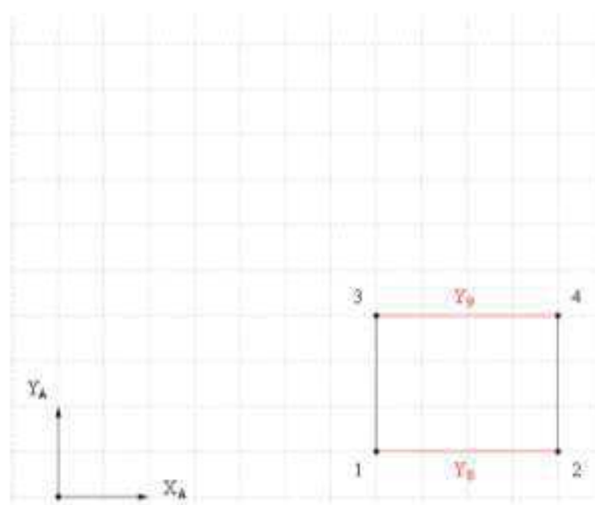


Figura 7. Base de soporte del robot durante la fase de balanceo de la extremidad izquierda.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

III. Determinación del balance estático durante la marcha

Se evalúan las posiciones articulares que definen un ciclo de marcha de la extremidad izquierda en el MGD del robot. Se calcula la posición del centro de masa del robot con respecto al sistema de coordenadas 0 y después, con respecto al sistema de referencia A. Luego, se determina la posición, con respecto al sistema de referencia A, de los puntos que definen la base de soporte del robot, y en general de todos los puntos de interés (fig. 8), durante las fases que componen el ciclo de marcha. Por último, se verifica que se cumpla la condición de balance estático en cualquier instante (fig. 9).

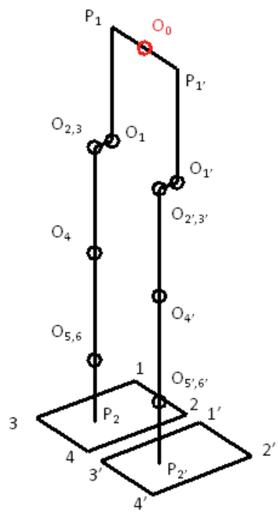


Figura 8. Puntos de interés del robot.

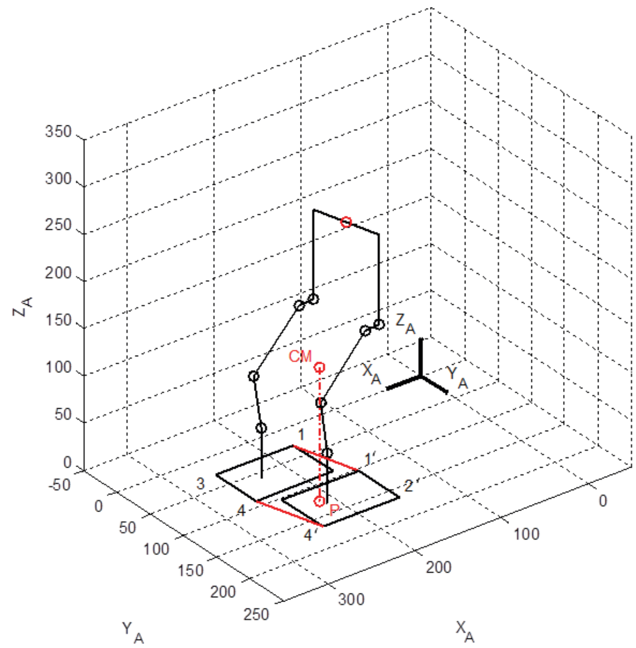
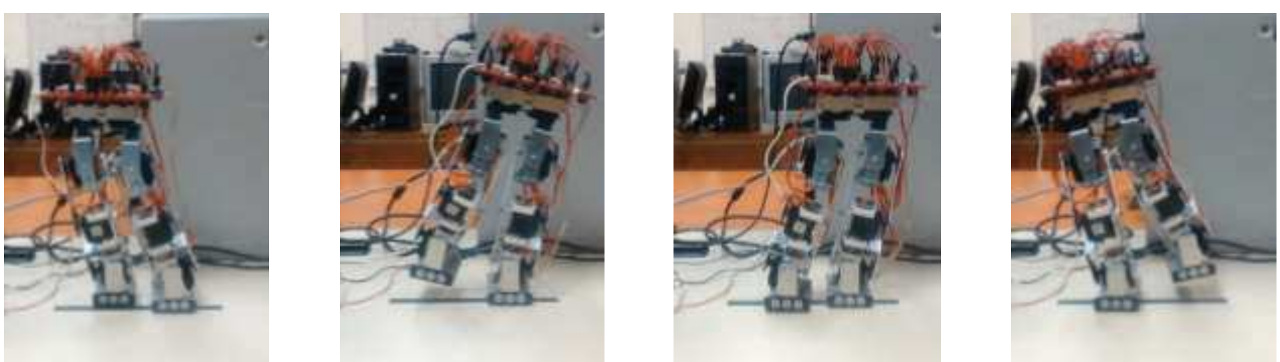


Figura 9. Proyección vertical del centro de masa del robot en un instante de la fase de segundo doble soporte de la extremidad izquierda. El punto P pertenece a la base de soporte del robot.

Los movimientos articulares que se evaluaron son lentos, por lo tanto, los efectos inerciales no son significativos, y el robot es capaz de ejecutar el ciclo de marcha al mantener el balance estático durante las 4 fases que lo definen (fig. 10), pues las únicas fuerzas que actúan sobre éste son su peso y la reacción del suelo.



(a) (b) (c) (d)

Figura 10. Robot bípedo en fases de (a) primer doble soporte, (b) soporte simple de la extremidad, (c) segundo doble soporte y (d) balanceo de la extremidad izquierda [1].

IV. Conclusión

Se describió una metodología para determinar el balance estático de un robot bípedo de forma analítica. La metodología se aplicó a un robot, cuya estructura mecánica incorpora 6 articulaciones rotacionales por cada extremidad, sin embargo, ésta es independiente del número y el tipo de articulaciones del robot, pues estos factores influyen únicamente en la obtención del MGD, por lo tanto, puede aplicarse a un robot que posea n grados de libertad por cada extremidad.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

La metodología descrita es útil para predecir la realización de un ciclo de marcha deseado, cuando los movimientos articulares que lo definen son lentos. En la siguiente etapa del proyecto a largo plazo del CICATA Querétaro, se realizará el análisis dinámico inverso de un robot que incorpore la distribución de grados de libertad que se muestra en la fig. 1, el cual consiste en la determinación de las fuerzas y los pares que actúan sobre los sistemas de coordenadas adjuntos a sus eslabones, cuando éste ejecuta un ciclo de marcha deseado, tomando en cuenta la velocidad y la aceleración angular con que se realiza cada movimiento articular; y se determinará el balance dinámico de éste durante la marcha sobre una superficie horizontal.

REFERENCIAS

[1] C. A. Matilde, "Diseño, análisis, simulación, construcción y programación de un robot bípedo de 12 grados de libertad, de marcha estáticamente balanceada, Tesis profesional, ITQ, 2016.
[2] J. J. Craig, Robótica. México: PEARSON, 2006, cap. 3.
[3] B. Siciliano y O. Khatib, Springer Handbook of Robotics. Berlin: Springer-Verlag, 2008, cap. 1.
[4] S. V. Shah, S. K. Saha, y J. K. D. Dynamics of Tree-Type Robotic Systems. New Yor: Springer, 2013, Cap. 4.
[5] M. Vukobratovic, B. Borovac y V. Potkonjak. "Towards a unified understanding of basic notions and terms in humanoid robotics", Robotica, 25, enero 2007, pag. 87-101.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 17, páginas 163 - 168.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Algoritmo de visión embebido para condiciones de iluminación no controladas

J. Contreras[1], J. Álvarez[2], and J. Herrera[3]

***Resumen*— Los algoritmos de visión requieren de un ambiente con parámetros de iluminación controlados y un sistema sofisticado para tener un desempeño adecuado, sin embargo, en el caso de las condiciones de iluminación no controladas y especificaciones de diseño compacto, se necesitan emplear otras técnicas para la detección de objetos. Para lo cual, en el presente trabajo, se desarrolló un algoritmo óptimo para el sistema embebido aplicado en condiciones no controladas. Se utilizó Matlab-Simulink, una cámara de visión de propósito general y una placa de desarrollo Raspberry Pi. Se caracterizó en diferentes condiciones de iluminación para determinar los valores de detección de objetos para las condiciones arbitrarias propuestas. El algoritmo propuesto cumple con las condiciones mínimas de cálculo para la información obtenida por el sistema embebido en la identificación de objetos. El sistema opera con un mínimo consumo de energía, en un tamaño compacto, en diversas condiciones de iluminación (que son útiles para las tareas de los robots de navegación), y se ha comprobado la detección del objeto en diferente tipo de escenario satisfaciendo el resultado estimado.**

I. INTRODUCCIÓN

Sucar [1] y González [6] concuerdan que un área muy ligada a la visión computacional es el procesamiento de imágenes, cuyo objetivo es extraer características de una imagen para su descripción e interpretación por medio de dispositivos como cámaras, laser, entre otros. Sin embargo, algunas aplicaciones no funcionan utilizando las técnicas conocidas, tal es el caso, en el ambiente debe ser controlado por dispositivos sofisticados y el tamaño de los equipos requeridos son muy grandes y poco portables.

Los sistemas que requieren obtener información sobre un objeto dado, consideran que la iluminación tiene un papel importante en esta práctica. Para ello se requiere una serie de métodos, mecanismos y dispositivos de iluminación que permiten el adecuado reconocimiento de objetos. Las aplicaciones no están diseñadas para funcionar sin un sistema de iluminación calibrado. A continuación se describen los procesos que intervienen en el reconocimiento de objetos en condiciones controladas de iluminación, en el cual se produce un alto consumo de energía y un elevado costo computacional/hardware [3].

Los procesos de iluminación integran diferentes tipos de iluminación (retroalimentación difusa, iluminación frontal, luz direccional, luz estructurada) y fuentes de iluminación (lámparas incandescentes, tubos fluorescentes, fibra óptica, laser). Con este sistema generado se podría obtener información del ambiente y reconstrucciones en 3D [4]. En cuanto a mecanismos autónomos la percepción que se le pueda dar, está definida por el tipo de sensores que lo incorpore para obtener información del ambiente [5]. En general, la iluminación es uno de los aspectos más importantes para reconocer objetos, ya que no existe un algoritmo capaz de reconocerlos en cualquier ambiente de iluminación [3].

Considerando las técnicas reportadas anteriormente que utilizan múltiples dispositivos, mecanismos, sistemas convencionales y software de procesamiento, no consideran el uso de arquitecturas más eficientes en cuestión de energía, tamaño, costo y capacidad de cómputo, por lo tanto, el objetivo del trabajo fue realizar un algoritmo de visión proponiendo otra solución de reconocimiento de objetos u obstáculos en un ambiente no controlado por medio de un sistema embebido que se comunica por medio de un adaptador WiFi hacía un modem con internet y éste a su vez se comunica a una computadora para el envío, recepción y procesamiento de las imágenes capturadas por una cámara conectada al mismo sistema embebido (figura 1).



Figura 1. Estructura de dispositivos y herramientas que intervienen para detectar objetos en condiciones no controladas.

Instituto Politécnico Nacional, Centro de Innovación y Desarrollo Tecnológico en Cómputo, Av. "Juan de Dios Bátiz" s/n esq. Miguel Othón de Mendizábal, Col. Nueva Industrial Vallejo, Del. Gustavo A. Madero, Ciudad de México, C.P. 07700.
E-mail: jesuscova11@gmail.com[1], jaalvarez@ipn.mx[2], jlozada@ipn.mx[3].

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

El presente trabajo aborda la implementación de una cámara web conectada mediante USB a una tarjeta de desarrollo Raspberry Pi, que ofrece los recursos necesarios a bajo costo. El utilizar estos dispositivos de éste tipo brindará las prestaciones que no se tienen con otras técnicas. El software utilizado es la herramienta MATLAB para la elaboración de éste sistema capaz de realizar el procesamiento de imágenes que se envía desde la Raspberry y poder visualizar los resultados en tiempo real de la detección de objetos.

La principal contribución del trabajo ha sido el desarrollo de un algoritmo simple que se encuentra embebido en un sistema de bajo consumo de energía, dimensiones reducidas y portabilidad que funciona sin ajustar el entorno y baja resolución. La función principal de la visión computacional es reconocer y localizar objetos mediante la información que pueda adquirir un dispositivo, utilizando procesamiento de imágenes y construir máquinas que puedan reconocer su entorno con capacidades similares a la visión humana. Existen numerosas aplicaciones las cuales dotan a un mecanismo con herramientas para captar el entorno y obtener información del mismo [1]:

- Robótica móvil y vehículos autónomos: se utilizan cámaras y otros tipos de sensores para localizar obstáculos, identificar objetos, encontrar caminos.
- Interpretación de imágenes áreas y de satélite.
- Análisis e interpretación de imágenes médicas, microscópicas, para astronomía, para comprensión, etc.

II. Formación y representación de la imagen

Cuando un sensor, una cámara o inclusive el ojo humano registran la radiación en el espectro visible, que ha interactuado en un ambiente con ciertos objetos físicos obteniendo información sobre su entorno formando imágenes, es decir, corresponde la intensidad o la brillantez (nivel de gris) en cada punto (pixel) de la imagen (asociada a una coordenada (x_i, y_i)). La imagen pude ser representada por una matriz f de dimensiones NxM [6], la función (1) es una representación matemática dado por los datos que se obtienen algún dispositivo y que se depositan en la matriz para realizar el análisis y procesamiento de las imágenes.

$$\boldsymbol{I_{i=0}}^{\boldsymbol{n}} = \boldsymbol{f}\,(\boldsymbol{x_i}, \boldsymbol{y_i}) = \begin{matrix} \boldsymbol{i_{0,0}} & \boldsymbol{i_{0,1}} & \boldsymbol{i_{0,m}} \\ \boldsymbol{i_{1,0}} & \ddots & \boldsymbol{i_{n-1,m}} \\ \boldsymbol{i_{n,0}} & \boldsymbol{i_{n,m-1}} & \boldsymbol{i_{n,m}} \end{matrix} \qquad (1)$$

Cada valor de intensidad le corresponde un nivel de gris, que va entre un número del 0 (negro) al 255 (blanco). Si se representan en 3 dimensiones, los dos valores primeros, corresponden a las coordenadas de la imagen (x_i, y_i) y el tercero representa el valor de intensidad (0-255) (figura 2).

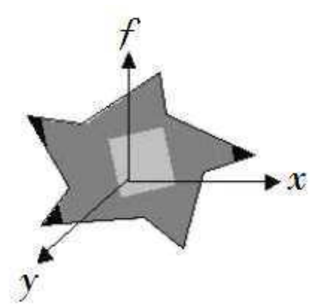


Figura 2. Representación de una imagen de 3 dimensiones.

El arreglo de las tres dimensiones es una representación de una imagen a color, ya que cada pixel de la imagen se especifica por 3 componentes RGB (Red, Green, Blue), uno para cada elemento (x_i, y_i). Éstos forman los colores primarios e indican los valores numéricos que se obtiene de la imagen, pueden ser expresados en un arreglo tridimensional proporcionado por (1):

$$\boldsymbol{f}\,(\boldsymbol{x_i}, \boldsymbol{y_i}, p) = (\boldsymbol{f_R}(x_i, y_i, 1), \boldsymbol{f_G}(x_i, y_i, 2), \boldsymbol{f_B}(x_i, y_i, 3)) \qquad (2)$$

Donde p representa el plano en que se está tomando la información, el valor 1 corresponde a la componente R, el valor 2 a la componente G y el valor 3 para la componente B. De esta manera se puede realizar procesamiento para cada color. En la figura 3, se muestra cómo se asignan valores en cada componente para formar un color específico, de igual manera los valores van de 0 (ausencia de color) a 255 (que representa que existe un color en su máxima intensidad).

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

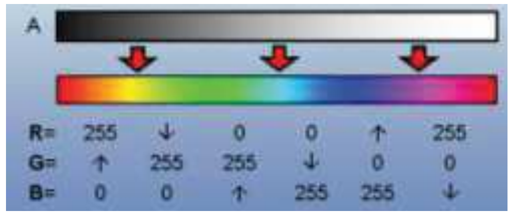


Figura 2.- Valores o niveles de brillantez cuando las imágenes sufren cambios de iluminación tanto para imágenes grises y a colores.

La iluminación es uno de los principales problemas de un mecanismo cuando se encuentra en su entorno, ya que con la variación de la misma se tienen muchas dificultades para la adquisición de la información [2]. Para lograr el reconocimiento de un objeto, las imágenes obtenidas se separan en unidades significativas, es decir, cada pixel de la imagen se va analizando, interpretando y modificando para obtener resultados con diferentes propósitos, a esto se le llama Segmentación.

Una de las formas que se puede segmentar una imagen es por medio de la determinación de los bordes. El problema radica en que las regiones son muy difíciles de encontrar ya que en ocasiones esos segmentos no se puede delimitar como unidades significativas. Las características más comunes para delimitar o segmentar regiones son: intensidad de los pixeles, textura, color, etc.

Uno de los problemas que ataca la segmentación es distinguir las variaciones propias del objeto o por cambios de iluminación (por ejemplo sombras), que podría tratarse de otro objeto. Existen 3 tipos de técnicas para la segmentación de regiones [1]:

- Locales: agrupan pixeles en base a sus atributos y de los de sus vecinos.
- Globales: se basan en propiedades globales de la imagen.
- División: combinan las técnicas anteriores.

III. Niveles de análisis y visión

La visión computacional se observa como un proceso de información, el cual se puede analizar de diversas formas, se propone 3 niveles [1]:

1. Teoría Computacional: metas y estrategias para analizar la información.
2. Representación y algoritmo: se describe el proceso del algoritmo para representar las entradas y salidas de variables.
3. Implementación: como se realiza fisicamente.

La visión consiste en obtener la información de una imagen y dividirla, ya que el proceso o los métodos son muy complejos, en cada etapa se va delimitando hasta llegar a la descripción deseada. Se conocen 3 niveles [1]:

- Procesamiento de nivel bajo: se utiliza para obtener contornos, gradiente, color, etc.
- Procesamiento de nivel intermedio: se agrupan elementos para obtener líneas, figuras poligonales, etc.
- Procesamiento de alto nivel: modelos o software especializado para el tratado de la información.

IV. Diseño del Algoritmo e Implementación

Hay diversos algoritmos que se pueden utilizar para reconocer objetos en un determinado ambiente natural, pero siendo un escenario diferente, con variación de la intensidad de luz, es muy difícil poder reconocerlos específicamente ya que su estructura de igual forma va cambiando de modo que lo detecta la cámara, el siguiente algoritmo tiene la finalidad de detectar los objetos con variación de la intensidad de luz.

La adquisición de los datos se obtiene por medio de una cámara web y se almacenan todos los valores existentes que pudiera conformar la escena en una matriz específicamente para el procesamiento correspondiente. El tamaño de la matriz (NxM) puede ser fijo o dinámico, ya que depende de la resolución de la cámara. Se recorre la matriz de izquierda a derecha y de arriba hacia abajo hasta recorrer todos los valores de la escena para separar los 3 componentes fundamentales dado por (2) se obtienen (3):

$$Para\ i = 0\ hasta\ N$$
$$Para\ j = 0\ hasta\ M \quad (3)$$

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

$$f_R\ (i,j,1) = matriz\ (i,j,1);$$
$$f_G\ (i,j,2) = matriz\ (i,j,2);$$
$$f_B(i,j,3) = matriz\ (i,j,3);$$

A partir de (3) podemos realizar el procesamiento que se requiera, tomando cada uno de los colores primarios por separado. Se requiere tener la imagen en escala de grises de manera que los objetos queden separados del fondo, aplicando una etiqueta a los componentes de mayor aproximación para extraer centros que estén presentes en la imagen. La escala de grises se obtiene sumando las 3 componentes y dividirlas entre 3 (4).

$$im_{gris} = (f_R(i,j,1) + f_G(i,j,2) + f_B(i,j,3))/3 \tag{4}$$

Posteriormente se requiere una imagen binaria con el fin de encontrar puntos en blanco y en negro para delimitar contornos o bordes. Para los bordes encontrados, limpiar el interior del objeto es una de las formas de detección, una forma de hacerlo es condicionando el valor de intensidad, si el valor del pixel supera el valor de intensidad definido (umbral). El pixel se le asigna un 0 (la imagen o la zona será de color negro), si no, el pixel se le asigna el valor 255 (color en blanco). El umbral permite tener un amplio margen de detección del objeto, ya que el límite que se le pueda dar, formará un objeto y parte de un fondo. Una vez encontrado los límites del área del objeto se establece la distancia para localizar un punto central de la imagen. Se sabe que la distancia se calcula con (5):

$$d = \sqrt{(x2 - x1)^2 + (y2 - y1)^2} \tag{5}$$

Posteriormente mide la región en donde se encuentre el objeto sobre componentes marcados con valores a 0, cada componente tendrá su elemento identificado. Realiza el mismo procedimiento hasta que detecte el centro del objeto. Operaciones con los ejes cartesianos con una distancia de 10 centímetros, que se le ha puesto como límite en los parámetros para la detección del objeto. Mostrará en pantalla el objeto detectado y la distancia que tiene sobre él.

La implementación de este tipo de algoritmo donde se utiliza la adquisición de imágenes en tiempo real, Matlab es un software muy eficaz ya que permite analizar datos por medio de matrices, realizar procesamiento de imágenes, crear modelos y aplicaciones complejas. Matlab incluye el paquete de Simulink, que ofrece la capacidad de programar hardware de bajo costo y que contiene una serie de herramientas para estos fines [7]. Una de ellas es el soporte de hardware para Raspberry Pi, ya que habilita la comunicación remota con Matlab. La Raspberry Pi es una computadora completa funcional de bajo costo basado en un procesador ARM11, que permite el desarrollo de diversos proyectos de enseñanza en diversas áreas [8]. Se utilizó una computadora convencional, donde se instaló Matlab versión 2015, una cámara web de propósito general, una Raspberry Pi 2 B+ con un adaptador WiFi para realizar la comunicación inalámbricamente (ya que también se puede realizar mediante cable Ethernet), un router para la conexión a internet y tarjeta de memoria SD de 8GB para instalar el sistema operativo de la Raspberry Pi.

Para realizar la comunicación entre la Raspberry Pi y Matlab, Matlab se encarga de proporcionar el sistema operativo de la Raspberry Pi (Raspbian) que a su vez tiene precargado herramientas para la comunicación con Simulink, que por medio de bloques permite la interacción con la información proveniente de la Raspberry Pi. Cuando Matlab termina de instalar el sistema operativo a la microSd con Raspbian, muestra 3 opciones para realizar la comunicación remota, localmente conectado al router, conectado directamente a la computadora, o manualmente ingresando un IP estática. En este caso se configuró la Raspberry Pi con una IP estática ya que la red es local. Inicialmente se configuraron la conexión de red tanto de la Raspberry Pi cómo de la computadora por medio de WiFi con la finalidad de estar conectados a la misma red. En la consola de Matlab se ingresa la siguiente instrucción para conectar la Raspberry Pi, se crea un objeto h con los siguientes datos:

$$h = raspberrypi('IPaddress'.'user\ name','password');$$

Se crea un objeto con el nombre de la variable h con los datos descritos en la instrucción. Confirmar que se ha conectado satisfactoriamente el enlace, se utiliza la siguiente instrucción:

$$message = h.connect$$

El modelo de la figura 3, fue realizado en Simulink, con bloques de procesamiento de imágenes.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

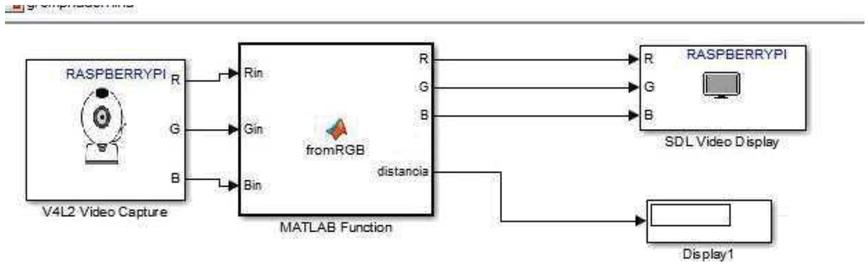


Figura 3. Modelo a bloques para el procesamiento de imágenes.

En la figura 3, se muestra como se realiza la captura de imágenes desde la Raspberry, con el bloque de Simulink, V4L2 Video Capture, el cual realiza la captura de video desde la cámara web USB en formato RGB con una resolución de *160 * 120* pixeles, constantemente envía información por medio de la conexión remota, Matlab recibe los datos en el siguiente bloque, Matlab Function, permite agregar código para realizar el procesamiento de las imágenes con su propio lenguaje de programación. En la función fromRGB se encuentra éste código donde se especifican las entradas (datos provenientes del bloque V4L2 Video Capture) y las salidas (variables para mostrar en pantalla o enviar a otro bloque). Como se muestra en la figura 3, las variables de entrada y que almacenan la información son: *Rin, Gin, Bin,* y las variables de salida son *R, G, B,* éstas se enviarán al bloque SDL Video Display para visualizar el resultado del procesamiento y la última salida *distancia,* se envía a un bloque Display 1 para visualizar la distancia que existe entre la cámara y el objeto.

Una función de Matlab que nos permite obtener en una sola matriz y utilizar los valores de una imagen a color, es decir, los 3 componentes, se utiliza *cat*:

$$img = cat\ (3, Rin, Gin, Bin);$$

Donde el valor 3 se especifica para matrices multidimensionales, *img* almacena los valores concatenados de las matrices. Se declaran las variables correspondientes a la resolución NxM, en este caso *160 * 120*, posteriormente se obtienen los valores de la escena recorriendo toda la imagen y aplicando el primer proceso, convertir la escena a escala de grises para obtener valores entre 0 y 255. Para convertir una imagen RGB a escala de grises, se debe sumar las 3 componentes y dividirlas entre 3. Aplicando (3) y (4) el código queda de la siguiente manera:

$$\begin{aligned} &Para\ i = 0\ hasta\ 160 \\ &\quad Para\ j = 0\ hasta\ 120 \\ &\qquad f_R(i,j,1) = (img\ (i,j,1) + img\ (i,j,2) + img\ (i,j,3))/3; \\ &\qquad f_G(i,j,2) = (img\ (i,j,1) + img\ (i,j,2) + img\ (i,j,3))/3; \\ &\qquad f_B(i,j,3) = (img\ (i,j,1) + img\ (i,j,2) + img\ (i,j,3))/3; \end{aligned} \tag{6}$$

Se comparan los valores obtenidos en el proceso anterior a un valor de intensidad determinado (umbral) para obtener una imagen binaria, en este caso el valor del umbral es de 80. Si el componente en su color es menor a 80 reemplazará esos valores colocando un 0 (negro) y si es mayor pondrá un valor de 255 (blanco). Se puede utilizar sólo un componente ya que todos los componentes tienen el mismo valor después de convertir la imagen a escala de grises.

$$\begin{aligned} &if\ \left(f_R(i,j,1)\right) > 80 \\ &\quad f_R(i,j,1) = 255; \\ &else \\ &\quad f_R(i,j,1) = 0; \end{aligned} \tag{7}$$

Conseguir la distancia entre el punto del objeto y la cámara se obtendrá por (5), Matlab proporciona la función *sqrt* para obtener la raíz de un valor (8).

$$distancia = sqrt((x_2 - x_1)^2 + (y_2 - y_1)^2) \tag{8}$$

Buscará los valores identificados con 0 para encontrar el contorno del objeto encontrado, si el objeto tiene valores dispersos dentro de su contorno, los pondrá a 0. La distancia la muestra en un bloque tipo Display para visualizar su valor en la simulación.

V. Resultados

Durante el desarrollo de este trabajo se realizaron diferentes pruebas las cuales llevaron a efectuar cambios en lo que se refiere a la resolución, ya que a mayor resolución de la cámara, es mayor el procesamiento que se consume en el sistema, considerando un retraso mayor a lo esperado. Por lo cual se elige de menor resolución para un mejor aprovechamiento de tiempo y costo computacional.

En la figura 4 se muestra el resultado experimental que se obtuvo, con el reconocimiento esperado para condiciones de iluminación no controladas, a fin de lograr el nivel de detección deseado. Se llevó acabo en un lugar cerrado a muy poca luz y con dos botellas de plástico, reconociendo éstos dos objetos por medio de su contorno.

De esta manera aunque el objeto cambie su apariencia debido a los cambios de iluminación, el sistema embebido será capaz de conocerlo hasta cierto punto. El umbral definido mostró un mejor resultado ya que el objeto detectado se muestra con mayor precisión del contorno.



Figura 4. Resultados del sistema embebido reconociendo objetos en condiciones no controladas.

VI. Conclusión

Con los resultados obtenidos se puede concluir que se ha conseguido crear un sistema embebido capaz de reconocer objetos en condiciones no controladas de iluminación, permitiendo la reducción de costos computacionales, el bajo consumo de energía y una mayor portabilidad del sistema. MATLAB y Simulink facilitan un entorno que permite acelerar la investigación, reducir los tiempos de análisis y desarrollo e implementar aplicaciones complejas en dispositivos de bajos recursos. Se demostró el uso de arquitecturas más eficientes como la Raspberry Pi para el procesamiento de imágenes logrando un mejor desempeño ante condiciones no ideales. Cómo trabajo a futuro se propone diseñar un prototipo y realizar la implementación de éste sistema embebido con la finalidad que sea autónomo, esquivando objetos de acuerdo a lo que capture y vaya detectando la cámara teniendo una distancia mínima para moverse a la derecha o a la izquierda. Ampliar las funcionalidades de Matlab-Simulink con bloques más complejos para la adquisición y procesamiento de imágenes de tal forma reducir código que pueda funcionar de forma independiente.

VII. Referencias

[1] L. Enrique Sucar, Giovani Gómez, “Visión Computacional,” Instituto Nacional de Astrofísica, Óptica y Electrónica, Puebla, México.

[2] J. Portillo. “Detección de movimiento de objetos mediante secuencias de video”. Instituto Politécnico Nacional. México, D.F., Mayo de 2012.

[3] E. García, C. Avilés,“Detección y clasificación de objetos dentro de un salón de clases empleando técnicas de procesamiento digital de imágenes”, México, Universidad Autónoma Metropolitana , 2008.

[4] E. A. Sobrado, “Sistema de visión artificial para el reconocimiento y manipulación de objetos utilizando un brazo robot,” Lima, Perú, Pontificia Universidad Católica del Perú, 2003.

[5] R. Arandes, “Implementación del sistea de localización y obtención del mapa de un robot móvil,” Madrid, Barcelona, Universidad Politécnica de Catalunha. 2015.

[6] A. González, F. Martínez, A. Pernía, F. Alba, “Técnicas y algoritmos Básicos de visión artificial,” Espana, Universidad de la Rioja, 2006 .

[7] The MathWorks, Inc. 1994-2016. Recuperado de: http://www.mathworks.com/index.html?s_tid=gn_logo.

[8] Raspberry Pi Foundation UK registered charity 1129409. Recuperado de: https://www.raspberrypi.org/

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 18, páginas 169 - 180.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Exoesqueleto para asistencia en terapias de rehabilitación de movimientos de los dedos de la mano

Aguilar-Pereyra J. Felipe[1], Castillo-Castañeda Eduardo[2]

[1] *Universidad Tecnológica de Querétaro, México, e-mail:* faguilar@uteq.edu.mx

[2] *Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada Querétaro, Instituto Politécnico Nacional, México, e-mail:* ecastillo@ipn.mx

Resumen — La demanda de servicios de terapia para rehabilitación del movimiento de las manos se ha incrementado en los últimos años. Se propone un exoesqueleto para movilizar los dedos de la mano en ejercicios de flexoextensión y oposición del dedo pulgar. El exoesqueleto para cada dedo puede ser ajustado a la longitud particular de los dedos del usuario. Los movimientos realizados por el exoesqueleto corresponden a la trayectoria natural de flexoextensión y oposición de los dedos de la mano para evitar dolor al usuario. La amplitud de la rotación de las articulaciones metacarpofalángica (MCP) e interfalángica proximal (PIP), tienen un intervalo de 0 to 90° y la articulación interfalángica distal (DIP) de 0 to 60°; las rotaciones de las articulaciones son coordinadas naturalmente. Los componentes del exoesqueleto son realizados mediante impresión 3D, lo que resulta en un dispositivo efectivo y de bajo costo. La personalización de las dimensiones del dispositivo permite cubrir un importante porcentaje de la población adulta.

I. INTRODUCCIÓN

En el estado de Querétaro no se cuenta con sistemas robóticos para auxiliar en la rehabilitación física de personas, ya que las terapias se aplican por terapeutas apoyados en sistemas de rehabilitación manuales. Por otra parte, se ha comprobado que el uso de sistemas robóticos en las terapias de rehabilitación incrementan los beneficios respecto a las terapias manuales al incorporar tareas de ejercicio intensivas e interactivas [1]. Aun cuando los sistemas robóticos no reemplazan a los terapeutas, constituyen una valiosa herramienta en el tratamiento de discapacidades [2].

El desarrollo de exoesqueletos para rehabilitación de la mano ha avanzado más lentamente que para otras partes del cuerpo debido en parte a las altas exigencias de capacidades de motorización y sensado [3]. *The rutgers master II* ha sido una de los primeros exoesqueletos para rehabilitación de mano; fue propuesto por Bouzit, Burdea, Popescu, y Boian en 2002 [4]. Éste es un exoesqueleto tipo guante que interacciona con un ambiente virtual 3D en tiempo real; se instrumentó usando actuadores neumáticos (servo válvulas neumáticas) y sensores de efecto hall para medir los ángulos de flexión y aducción, Figura1.



Figura 1 The rutgers master II [4]

También enfocados en la rehabilitación, [5] desarrollaron un exoesqueleto para la rehabilitación de la mano, empezando con la construcción de un prototipo mecánico de cuatro grados de libertad, el cual es movido por una unidad actuadora con motor de CD y reductor de velocidad.

En Figura 2 se muestra el exoesqueleto desarrollados por Wege y Zimmermann. Un problema presente en los exoesqueletos es que resulta muy complejo modificar sus dimensiones para adaptarlos al tamaño de las manos de los usuarios. Además, la masa del exoesqueleto debe ser cargado por la mano del usuario, lo que pronto puede provocar fatiga y reducir el tiempo de ejercicio.

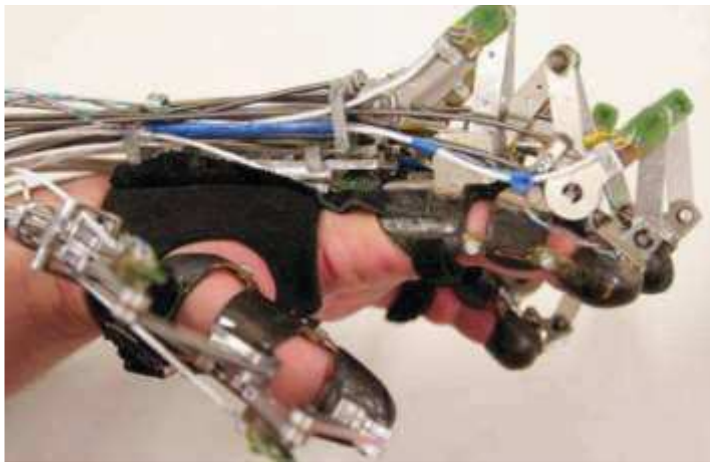


Figura 2 Exoesqueleto para la rehabilitación de la mano [6]

II. MODELO CINEMÁTICO DE LA MANO

El modelado cinemático de la mano es un problema complejo, para atender este problema se han aplicado técnicas de la robótica [7]. El planteamiento consiste en considerar la mano humana como una cadena cinemática ramificada. La mano es claramente el conjunto articulado más complejo del esqueleto humano, jerárquicamente puede ser establecido en dos grupos cinemáticos: (i) los dedos largos y el pulgar y (ii) los huesos del carpo [7]. Se puede plantear un modelo simplificado si se considera a la palma de la mano como una base fija y a cada uno de los dedos como una cadena cinemática abierta, Figura 3.

Un planteamiento realista para realizar la flexoextensión de los dedos en terapias de rehabilitación puede considerar para cada uno de los dedos, índice a meñique, una cadena cinemática de cuatro eslabones y cinco articulaciones a partir de la muñeca [8]. La ecuaciones del modelo cinemático pueden calcularse por medio de los parámetros de Denavit-Hartenberg (DH), esta convención se utiliza comúnmente en el modelado de mecanismos y de robótica [9]. La ecuaciones del modelo cinemático pueden calcularse por medio de los parámetros de Denavit-Hartenberg (DH), esta convención se utiliza comúnmente en el modelado de mecanismos y de robótica [9].

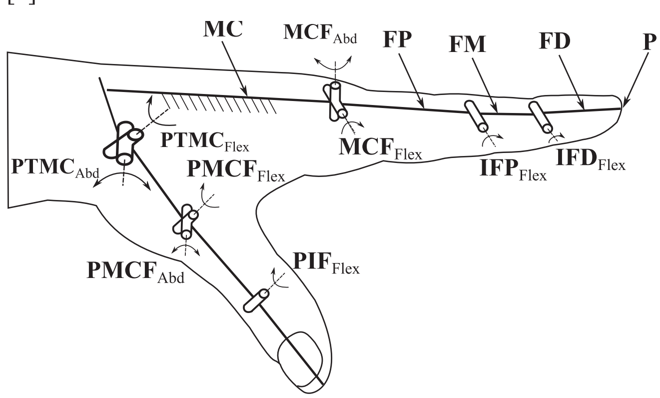


Figura 3 Modelo cinemático de la mano considerando la palma fija

En la Tabla 1 se muestran los parámetros de DH para el dedo pulgar propuesta por [8]. La Tabla 2 presenta los parámetros de DH para los dedos largos: índice, medio, anular y meñique.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 18, páginas 169 - 180.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Tabla 1 Parámetros de DH para el pulgar

Articulación	Θ_i	d_i	a_i	α_i
1	$\Theta_{TMC\ ab/ad}$	0	0	$\pi/2$
2	$\Theta_{TMC\ f/e}$	0	L_1	0
3	$\Theta_{MCF\ f/e}$	0	L_2	0
4	$\Theta_{IFP\ f/e}$	0	L_3	0

Dónde: L_1, L_2 y L_3 son las longitudes del metacarpiano y las falanges proximal y distal del pulgar respectivamente. La flexoextensión de los dedos largos ocurre en un plano distinto para cada dedo; además, la abducción es nula en flexión completa pero alcanza su movimiento máximo en extensión completa [10].

Tabla 2 Parámetros de DH para los dedos largos

Articulación	Θ_i	d_i	a_i	α_i
1	Θ_{CMC}	0	L_4	$\pi/2$
2	$\Theta_{MCF\ ab/ad}$	0	0	$-\pi/2$
3	$\Theta_{MCF\ f/e}$	0	L_5	0
4	Θ_{IFP}	0	L_6	0
5	Θ_{IFD}	0	L_7	0

L_4, L_5, L_6 y L_7, son las longitudes del metacarpiano y las falanges proximal, medial y distal de los dedos largos respectivamente.

En Figura 4 se ilustra la flexión del dedo índice con movimiento de flexión de las articulaciones metacarpofalángica e interfalángicas de 0 a 90° con origen en la articulación metacarpofalángica.

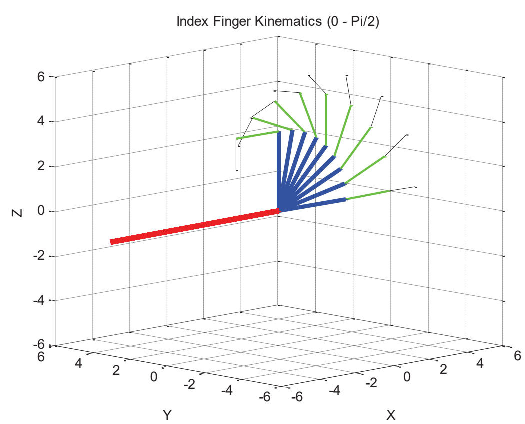


Figura 4 Simulación de flexión del dedo índice

La movilidad del modelo cinemático ramificado es determinado considerando todas las articulaciones independientes, sin embargo, el sistema nervioso central coordina los movimientos de las articulaciones para la realización de distintos tipos de tareas creando sinergias. Estas sinergias existen de manera natural al manipular objetos o utilizar herramientas.

Por otra parte, existen restricciones de movimiento de la mano están divididas en dos grandes grupos: restricciones estáticas y restricciones dinámicas debidas a las dependencias entre ángulos de las articulaciones [11].

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Tabla 3 Restricciones estáticas para el pulgar

Dedo / Articulación	*Identificador*	*Flexión*	*Extensión*	*Abducción / aducción*
Pulgar	1			
Trapecio-metacarpiana	TMC	50°- 90°	15°	45°- 60°
Metacapofalángica	MCF	75° - 80°	0°	0°
Interfalángica	IF	75° - 80°	5° - 10°	0°

Las restricciones estáticas se refieren a los límites de movimiento de las articulaciones y han sido establecidos valores estándar en la fisiología articular (Kapandji, 2006), Tablas 3 y 4. Las restricciones dinámicas son debidas a la relación de movimiento entre articulaciones; una de las más importantes es la relación entre las articulaciones IFD e IFP para los cuatro dedos largos, está establecida por ec. (1) [12] [13].

$$\Theta_{IFD} = \left(\frac{2}{3}\right)\Theta_{IFP} \tag{1}$$

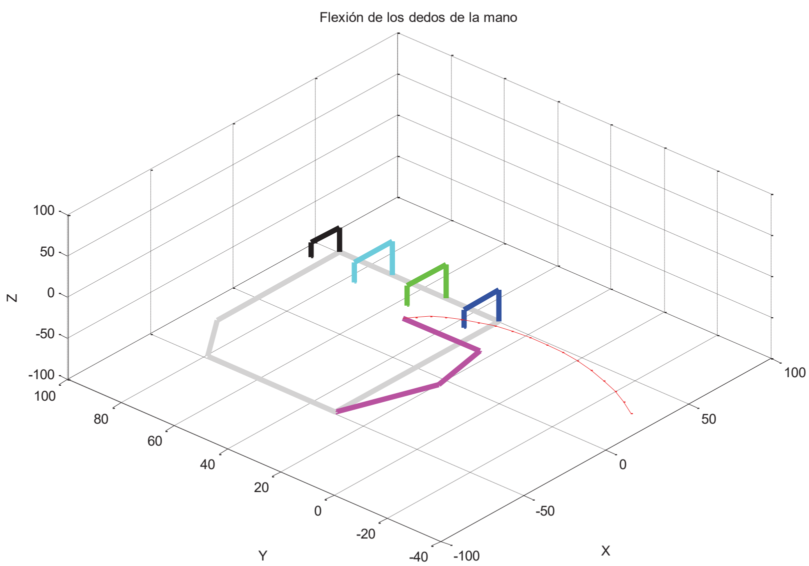


Figura 3 Simulación de movimientos de flexión de los dedos y oposición del pulgar

III. EXOESQUELETO PROPUESTO

Para satisfacer los requerimientos de flexoextensión de los dedos largos y oposición del pulgar, se ha propuesto un guante tipo exoesqueleto accionado por cables a manera de tendones. Inspirado en el accionamiento natural de los dedos, en este mecanismo se tienen dos tendones artificiales, uno flexor y otro extensor por cada dedo.

La estrategia seleccionada consiste en fabricar los componentes de los segmentos del exoesqueleto a la medida de las dimensiones de los dedos de la mano que se desea rehabilitar. Esta fabricación se simplifica con la disponibilidad de imprimir en tres dimensiones las piezas en plástico Poliácido Láctico (PLA), el cual es un polímero biodegradable proveniente del almidón del maíz, suficientemente rígido y resistente para este propósito. Además, el costo de fabricación de las piezas correspondientes a los dedos de una mano incluyendo el pulgar es aproximadamente un mil quinientos pesos mexicanos, lo que resulta en un costo viable para la personalización del exoesqueleto para cada paciente. El tiempo de impresión de los componentes del exoesqueleto permite construirlo en tiempo promedio de un día.

Tabla 4 Restricciones estáticas para los dedos largos [7][8]

Dedo / Articulación	*Identificador*	*Flexión*	*Extensión*	*Abducción / aducción*
Índice	2			
Carpometacarpal	CMC	5°	0°	0°
Metacapofalángica	MCF	90°	30°- 40°	60°
Interfalángica Proximal	IFP	110°	0°	0°
Interfalángica Distal	IFD	80° - 90°	5°	0°
Medio	3			
Carpometacarpal	CMC	5°	0°	0°
Metacapofalángica	MCF	90°	30°- 40°	45°
Interfalángica Proximal	IFP	110°	0°	0°
Interfalángica Distal	IFD	80° - 90°	5°	0°
Anular	4			
Carpometacarpal	CMC	10°	0°	0°
Metacapofalángica	MCF	90°	30°- 40°	45°
Interfalángica Proximal	IFP	120°	0°	0°
Interfalángica Distal	IFD	80° - 90°	5°	0°
Meñique	5			
Carpometacarpal	CMC	15°	0°	0°
Metacapofalángica	MCF	90°	30°- 40°	50°
Interfalángica Proximal	IFP	135°	0°	0°
Interfalángica Distal	IFD	90°	5°	0°

La fabricación de los tres componentes del exoesqueleto de un dedo, proximal, medial y distal, dependen básicamente de tres dimensiones: longitud (L), altura (H) y anchura (W) de cada segmento del dedo. Otra característica de los segmentos del exoesqueleto es que son muy ligeros, menos de diez gramos por cada componente, lo que resulta cómodo para el usuario. Para el diseño y construcción de un prototipo se consideran las dimensiones de una mano en particular de la Tabla 5.

Tabla 5a Dimensiones de una mano en particular para el diseño del exoesqueleto

Dedo	**Pulgar**		
Segmento	L	H	W
Proximal (mm)	28.48	24	23
Medial (mm)	N. A.	N. A.	N. A.
Distal (mm)	27.56	20	22
Dedo	**Índice**		
Segmento	L	H	W
Proximal (mm)	37.67	24	24
Medial (mm)	21.13	16	21
Distal (mm)	21.59	15.5	18
Dedo	**Medio**		
Segmento	L	H	W
Proximal (mm)	41.8	24	24
Medial (mm)	25.72	18	21
Distal (mm)	22.51	16	17.5

Tabla 5b Dimensiones de una mano en particular para el diseño del exoesqueleto

Dedo	**Anular**		
Segmento	L	H	W
Proximal (mm)	30.04	21	23
Medial (mm)	23.89	17	20
Distal (mm)	23.43	15	17
Dedo	**Meñique**		
Segmento	L	H	W
Proximal (mm)	31.23	19	20.5
Medial (mm)	16.54	14	19
Distal (mm)	20.21	13.5	16

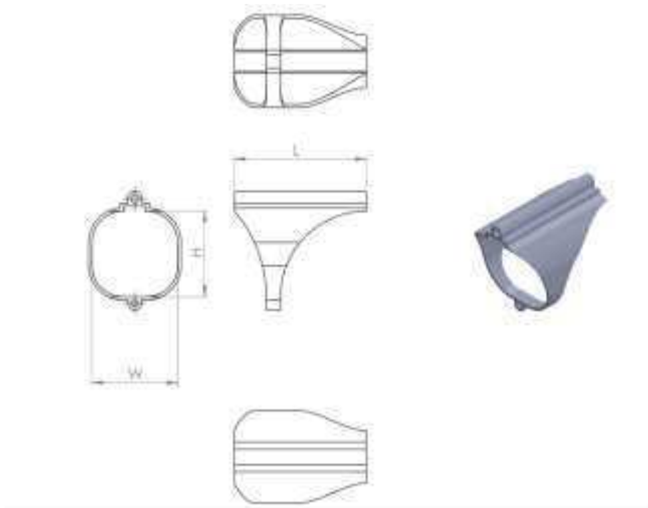


Figura 6 Componente IEXOP correspondiente al segmento proximal del dedo

Se ha diseñado cada segmento del exoesqueleto para su correspondiente segmento del dedo, el primero tiene dos guías por las cuales pasan los cables de los tendones flexor y extensor. El tendón extensor es un cable de acero de 1.66 mm (1/16 pulgada) de diámetro, el cual por su grosor sirve de guía a todo el exoesqueleto para evitar que se flexionen de manera lateral las articulaciones IFP e IFD. En Figura 6 se muestra el componente denominado IEXOP debido a que corresponde al segmento proximal del dedo índice. Las figuras 7 y 8 muestran los componentes denominados IEXOM e IEXOD debido a que corresponde al segmento proximal del dedo índice. La Figura 9 ilustra el Ensamble del exoesqueleto para el dedo índice.

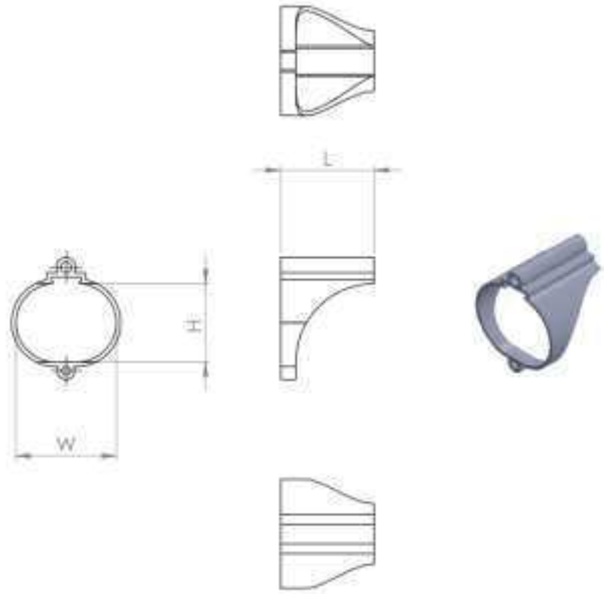


Figura 7 Componente IEXOM correspondiente al segmento medial del dedo

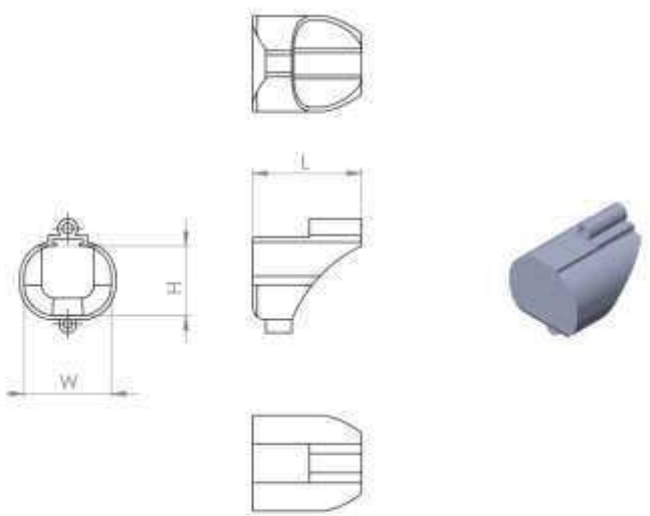


Figura 8 Componente IEXOD correspondiente al segmento distal del dedo

Las secciones rígidas del exoesqueleto sujetan los segmentos del dedo permitiendo la rotación natural de las articulaciones por la fuerza ejercida en puntos específicos del exoesqueleto. Los tendones artificiales están ubicados a lo largo del dedo en la parte dorsal media de tal manera que provocan una alineación natural de las falanges. Para encontrar un modelo simplificado pero funcional de los dedos, se considera cada uno de ellos como una cadena cinemática abierta de cuatro grados de libertad con la base en la palma de la mano, la cual es considerada como el eslabón 1 (fijo). La Figura 10 muestra el modelo propuesto que considera sólo tres GDL para realizar la flexoextensión. En esta simplificación del modelo del dedo, cada uno de los eslabones es considerado como un cuerpo rígido de longitud fija y sección transversal circular, su centro de masas *CGi* se establece en la parte central del eslabón. El eje de rotación de la primera articulación se asigna al punto O, el ángulo de rotación es $\Theta 2$, el punto A representa el extremo del eslabón 2. El peso del eslabón 2 se representa por la fuerza g2, la cual actúa siempre en dirección de la aceleración de la gravedad. Se asume que el torque T12 produce el movimiento de flexo-extensión del eslabón 2 (FP), en ausencia de este torque la fuerza Fp2

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

produce el mismo movimiento.



Figura 9 Ensamble del exoesqueleto para el dedo índice

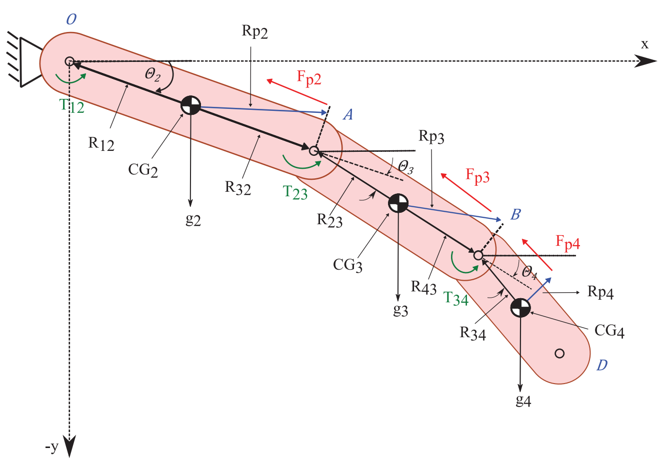


Figura 10 Modelo del dedo como una cadena cinemática de tres GDL

En la flexión del dedo medio, que es el más largo, se tiene el mayor incremento en la longitud del tendón flexor, que es también la carrera del actuador lineal.

El tendón artificial tendrá una longitud inicial:

$$L_{ini} = L_2 + L_3 + \frac{L_4}{2} \tag{2}$$

El incremento en la longitud L_2 debida a la rotación de la articulación MCF es

$$\Delta L_2 = \frac{2 * \pi * \frac{H_2}{2}}{4} = \frac{\pi * H_2}{4} \tag{3}$$

Para $0.5H_2$=19.25 mm, ΔL_2 = 15.11 mm; $0.5H_3$=15.25 mm, ΔL_3 = 11.97 mm; $0.5H_4$=14.25 mm, ΔL_4 = 11.19 mm;

La carrera del actuador lineal es

$$\Delta L_{tot} = \Delta L_2 + \Delta L_3 + \Delta L_4 \tag{4}$$

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

$\Delta L_{tot} = 15.11 + 11.97 + 11.19 = 38.28$ mm

Por lo anterior, se ha seleccionado el actuador de la marca Firgelli ® modelo L12 50 100 12 I, En Figura 11 se muestra el actuador seleccionado.



Figura 11 Actuador lineal eléctrico

IV. RESULTADOS

El prototipo construido del exoesqueleto propuesto se muestra en Figura 12, se pueden apreciar los segmentos correspondientes a las falanges de los cinco dedos.

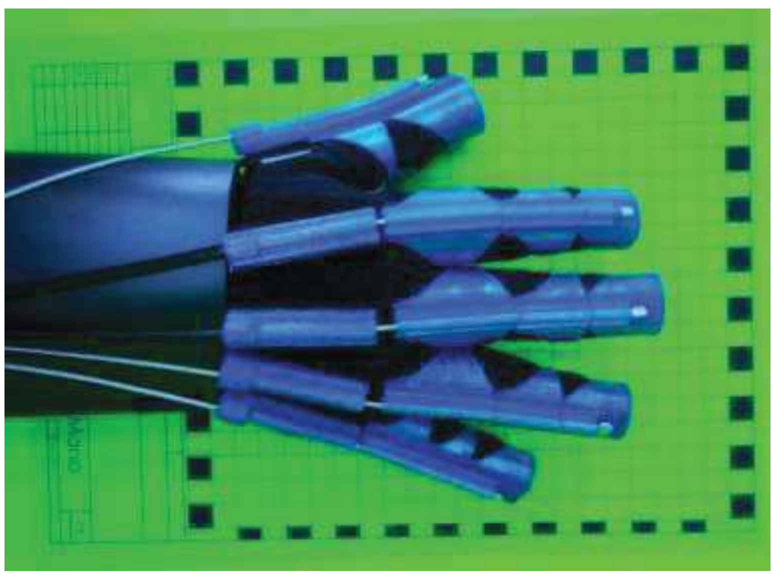


Figura 12 Guante tipo exoesqueleto propuesto

En Figura 13 se muestran tres imágenes de una secuencia de flexión de los dedos largos.

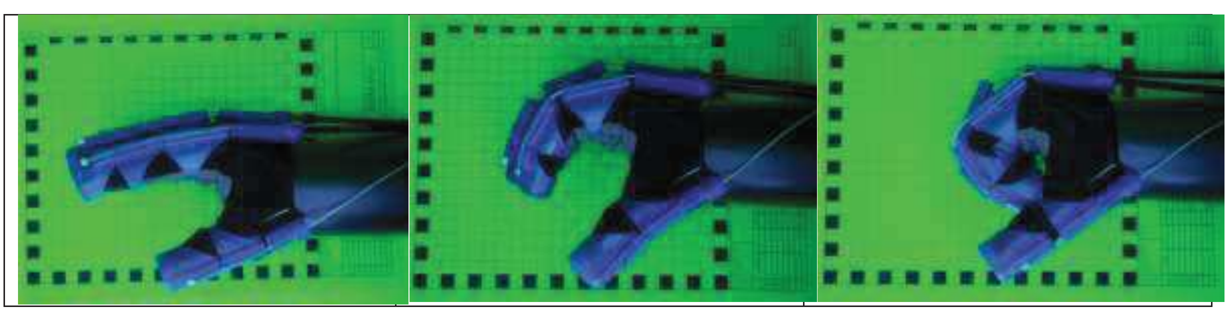


Figura 13 Flexión de los dedos largos

Los resultados de la simulación cinemática del modelo del dedo como una cadena cinemática de tres GDL, correspondiente a la figura 10, se muestran en las figuras 14, punto A, y 15 punto D.

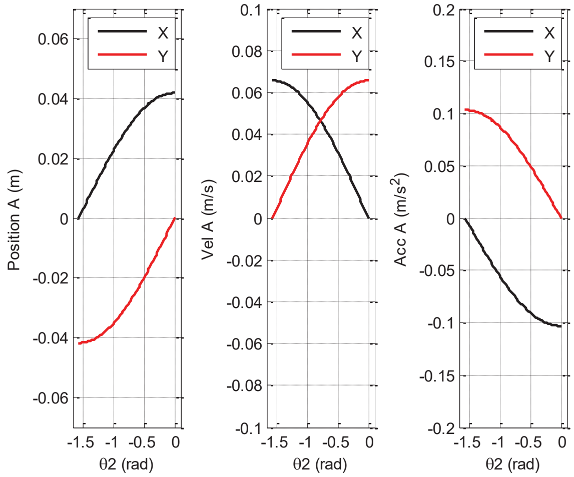


Figura 14 Posición, velocidad y aceleración del punto A

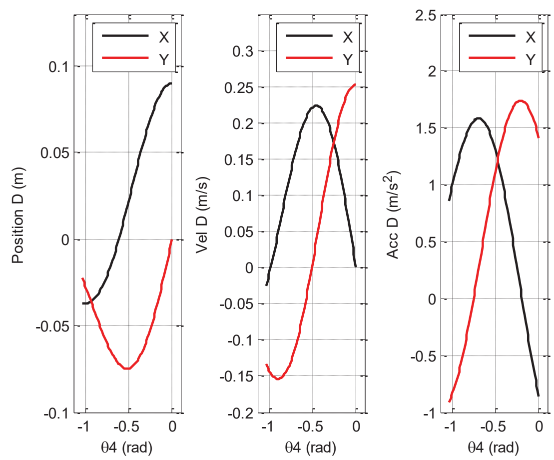


Figura 15 Posición, velocidad y aceleración del punto D

V. CONCLUSIÓN

A partir de las primeras visitas al Centro de Rehabilitación Integral de Querétaro se han identificado áreas de oportunidad para el desarrollo de proyectos de investigación de ciencia aplicada y tecnología avanzada de gran necesidad en el sector salud de nuestro país. En particular se identificó la necesidad de desarrollar un sistema de rehabilitación de mano ya que actualmente las movilizaciones de los dedos de la mano las realizan los terapeutas de forma manual. Por otra parte, se ha revisado la biomecánica y cinemática directa de la mano, lo cual ha permitido contar con el conocimiento necesario para la conceptualización de dos sistemas de rehabilitación para la mano, uno flexible tipo exoesqueleto.

A partir de la construcción de prototipos que se desarrolla actualmente y su evaluación, se podrá seleccionar el más adecuado para su construcción definitiva. El trabajo a desarrollar de forma inmediata consiste en la

conclusión de la construcción del exoesqueleto y su evaluación funcional. A mediano plazo se contempla el diseño y construcción del controlador correspondiente, así como la instrumentación necesaria para la medición de fuerza en los tendones artificiales y flexión de las articulaciones.

RECONOCIMIENTOS

Los autores agradecen el apoyo del CICATA Querétaro y de la Universidad Tecnológica de Querétaro por el apoyo brindado a esta investigación.

Los autores agradecen el apoyo financiero por parte del CONACYT. Los autores también agradecen la ayuda de los terapeutas del Centro de Rehabilitación Integral de Querétaro por su apoyo en la revisión de los protocolos de rehabilitación.

REFERENCIAS

[1] Burgar, C. G., Lum, P. S., Shor, P. C., & Machiel Van der Loos, H. F. (2000). Development of robots for rehabilitation therapy: the Palo Alto VA/Stanford experience. Journal of Rehabilitation Research and Development, 37(6), 663–73. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/11321002

[2] Levanon, Y. (2013). The advantages and disadvantages of using high technology in hand rehabilitation. Journal of Hand Therapy : Official Journal of the American Society of Hand Therapists, 26(2), 179–83. http://doi.org/10.1016/j.jht.2013.02.002

[3] Heo, P., Gu, G. M., Lee, S., Rhee, K., & Kim, J. (2012). Current hand exoskeleton technologies for rehabilitation and assistive engineering. International Journal of Precision Engineering and Manufacturing, 13(5), 807–824. http://doi.org/10.1007/s12541-012-0107-2

[4] Bouzit, M., Burdea, G., Popescu, G., & Boian, R. (2002). The Rutgers Master II-new design force-feedback glove. … /ASME Transactions on, 7(2), 256–263. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1011262

[5] Wege, A., & Hommel, G. (2005). Development and control of a hand exoskeleton for rehabilitation of hand injuries. … Robots and Systems, 2005.(IROS 2005). …, (1). Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1545506

[6] Wege, A., & Zimmermann, A. (2008). Electromyography Sensor Based Control for a Hand Exoskeleton, 1470–1475.

[7] Tondu, B. (2009). Kinematic modelling of anthropomorphic robot upper limb with human-like hands. Advanced Robotics, 2009. ICAR 2009. International Tondu, B. (2009).

[8] Cobos, S., Ferre, M., & Uran, S. (2008). Efficient human hand kinematics for manipulation tasks. In International Conference on Intelligent Robots and Systems (pp. 22–26). Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4651053

[9] Craig Jhon J., (2004) Introduction to Robotics: Mechanics and Control, . Addison-Wesley Longman Publishing Co. Inc., Third Edition, Boston, MA, USA.

[10] Kapandji, A. I. (2006). Fisiología articular: esquemas comentados de mecánica humana. (Editorial Médica Panamericana, Ed.) (6th ed.). Madrid.

[11] Erol, A., Bebis, G., Nicolescu, M., Boyle, R. D., & Twombly, X. (2007). Vision-based hand pose estimation: A review. *Computer Vision and Image Understanding*, *108*(1-2), 52–73. http://doi.org/10.1016/j.cviu.2006.10.012

[12] Lee, S. W., Chen, H., Towles, J. D., & Kamper, D. G. (2008). Estimation of the effective static moment arms of the tendons in the index finger extensor mechanism. *Journal of Biomechanics*, *41*(7), 1567–73. http://doi.org/10.1016/j.jbiomech.2008.02.008

[13] Kuch, J. J., & Huang, T. S. (1994). Human computer interaction via the human hand: a hand model. *... , 1994. 1994 Conference Record of the ...*, *1*, 1252–1256. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=471659

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 19, páginas 181 - 191.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Reference trajectory generation for a knee assistive device

B. Chaparro-Rico[1], E. Castillo-Castaneda[1]

[1]*National Polytechnic Institute –CICATA, Qro., Mexico, e-mail: betsychaparro@hotmail.com, ecast63@yahoo.com*

Abstract. The assistive devices for physical therapy require reference trajectories to provide personalized service, according to the physical requirements of each patient and according to the medical diagnostic. This paper presents a method to generated reference trajectories for a personalized therapy in a knee assistive device. The method automatically selects the order of polynomials to fit a set of trajectories performed by the patient. The polynomials are calculated by using regression analysis. The method is applied by using a knee assistive device; several therapist demonstrations of four knee exercises are acquired and processed to obtain the reference trajectories. In addition, the demonstration that fits better all the stored trajectories is selected.

Keywords: Reference trajectories, regression analysis, medical robotics, physical therapy.

I. INTRODUCTION

Mechanical Rehabilitation therapy varies for each patient according to the physical characteristics, the type of injury or illness, and the medical diagnostic, among others. Therefore, a physical rehabilitation device should reproduce a different trajectory for each patient. However, many assistive devices for limb rehabilitation only reproduce predefined trajectories (Refs.[1, 2]) without ability to adapt to the therapy of each patient. On the other hand, some of the most advanced limb rehabilitation robots remember the positions of a movement demonstrated by the therapist and after they reproduce the remembered positions, however, they reproduce the exact recorded trajectory from just one sample. Among these, ARMIN (Ref. [3]) for upper limb rehabilitation and Physiotherabot (Ref. [4]) for lower limb rehabilitation. However, human motion can't be represented by a single demo because it may contain errors due to: interaction between therapist, patient and robot; changes in speed and acceleration; and possible therapist hesitation. The movement should be learned by processing several demonstrations of the movement (Ref. [5]).

The rehabilitation robotics, considering trajectory generation, has focused on the characterization of the normal gait, with applications principally in correcting posture by exoskeletons, and the development of adaptive controls for reproducing trajectories (Refs. [6, 7]). Generation and selection of human movement trajectories have been studied more extensively in humanoid robotics, specifically in learning by imitation, learning by demonstration or programming by demonstration (Refs. [8, 9]).

In this paper, trajectories of knee exercises are stored by using knee assistive device and reference trajectories are successfully generated by regression. The proposed method provides greater flexibility to the therapy responding to specific requirements for each patient. Since regression analysis method fits the curves by using an order polynomial previously selected by the user, the technical relevance of this paper is that the order polynomial is automatically selected by evaluation of the slopes of the error curves.

II. EXERCISE DEMONSTRATIONS BY USING THE ASSISTIVE DEVICE

A knee assistive device, developed at IPN-CICATA, QRO., has been used for the experimentation. The mechanism is based on a five-bar mechanism with two degrees of freedom (Ref. [10]). The end effector of the mechanism supports the ankle and guides the patient's leg while a rehabilitation exercise is reproduced; it can reproduce several trajectories inside its work space. The device has encoders (position sensors) on its joints R_1 and R_2 which are fixed to the mechanism base, where the P_1 and P_2 are the prismatic joints and each R_i is a rotational joint, Fig. 1.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

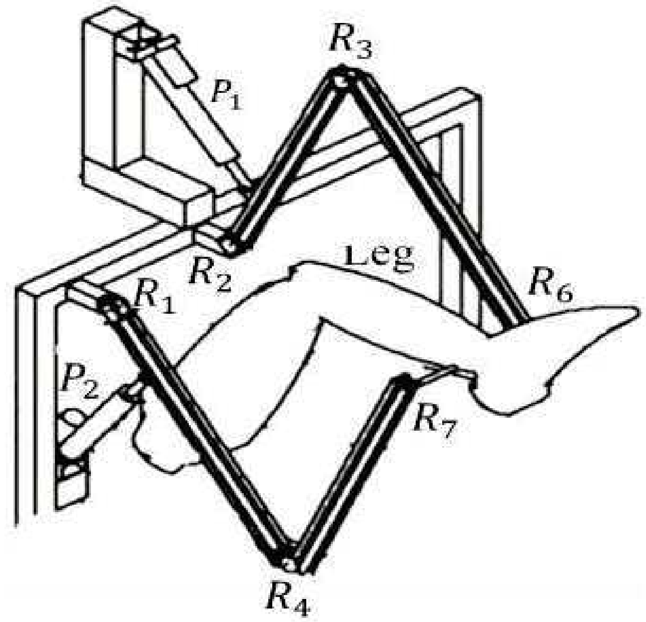


Fig. 1 Scheme of the knee rehabilitation device

Four exercises have been performed on the device while joints positions R_1 and R_2 have been sampled every 25ms. Sets of ten trajectories has been obtained for each exercise. Using the joints positions of R_1 and R_2, the end effector mechanism trajectories (X, Y) has been calculated by means of the forward kinematic. Every trajectory stored by the device has been also named "Demonstration". The exercises have been advised by specialists from CRIQ (Integral Rehabilitation Center of Queretaro). Fig. 2 shows the four knee rehabilitation exercises performed by a therapist and Fig. 3 shows a picture of the final prototype of the knee assistive device while an exercise is demonstrated with a person.

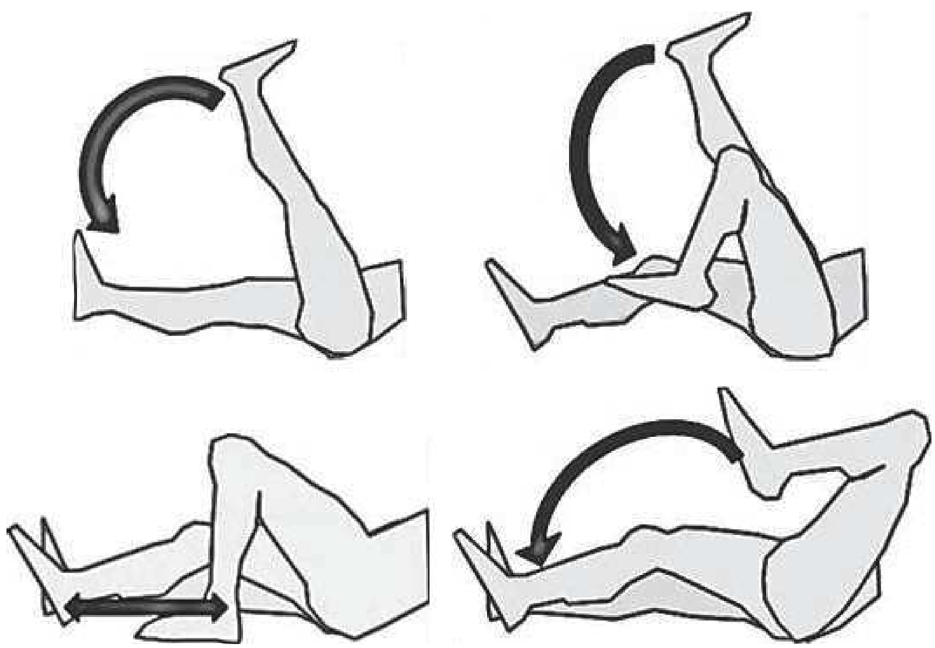


Fig. 2 Knee exercises recommended by the therapist.



Fig. 3 Knee assistive device

In this stage, the trajectories of four knee exercises have been stored by the device as example to apply the method. However, the trajectories should be updated in each therapy session by using samples from each patient. Therefore, the therapist can teach the desired trajectories to the assistive device.

III. AUTOMATIC SELECTION OF THE ORDER POLYNOMIAL FOR CURVE GENERATION BY REGRESSION

The regression method founds the coefficients of a polynomial with a known order polynomial under the minimum error criterion (Eq. 1). Where $\hat{b}$ is the generated values for the evaluated Cartesian component (X or Y); b is the valuated Cartesian component; i is the position array and k is the length of the data.

$$E = \sum_{i=1}^{k} \left(\hat{b} - b_i\right)^2 \quad (1)$$

The regression method calculates the coefficients polynomials which best fit a curve but the order polynomials should be selected by the user. Thus, in order to automatically select the order polynomial *N* and M, for the Cartesian Components *X* and *Y* respectively, the error curves have been evaluated (Refs. [11-14]), from residues between the input data and the predicted data. The input data contains the values of the trajectories stored by the assistive device (demonstrations) and the output data contains the values of the reference trajectory generated by regression. The residual distance r_i formed between the input curve (x, y) and the predicted curve $(\hat{x}, \hat{y})$ is given by,

$$r_i = \sqrt{(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2} \quad (2)$$

And the error for each evaluated demonstration is given by,

$$eXY_d = \sqrt{\frac{\sum_{i=1}^{k} {r_i}^2}{k}} \quad (3)$$

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Where d is the evaluated demonstration. The average error is given by,

$$eXYP = \frac{\sum_{d=1}^{D} eXY_d}{D} \quad (4)$$

Where D is the total of evaluated demonstrations.

The proposed method varies the order polynomial for both X axis and Y axis data, permuting different orders by scanning in different ranges from 1 to 1, 1 to 2, ..., 1 to 20, each scanning evaluates the permutations from de minimum value of the range to maximum value of the range. Then, the polynomial permutation with the smallest error is selected in each scanning (to reduce the data to a select group of 20 permutations) and a curve is generated displaying the error behavior. The method was applied for several sets of demonstrations for the four exercises.

According to the normalized error curves, the error value decreases while the order of the polynomials increases; until that the error reduction is not appreciable. When the error reduction is not appreciable, the polynomials with high order are not useful. In order to detect when the error reduction is not appreciable, the slope of the error curve has been used to select the best permutation polynomial. The curve slope “m”, of Fig. 4, can be calculated as:

$$m = \left|(eXYP_{(n+1)} - eXYP_{(n)})/((s_{n+1}) - (s_n))\right| \quad (5)$$

Where, *m* is the slope of curve; *s* is the scan number and *n* is the position array.

When m<1.0 for first time, s_n is selected. Every s_n has a preselected permutation polynomial with the minimum $eXYP_{(n)}$, then both sn and its corresponding permutation polynomial are automatically selected. As example, from the exercise A, Table 1 shows each scanning “s_n” with its corresponding evaluation range and the selected permutation polynomial exhibiting smallest *eXYP*.

Table 1 *eXYP*- exercise A

s_n	range	Permutation exhibiting smallest *eXYP*		Normalized *eXYP*	m
		N	M		
1	1 to 1	1	1	100	51.9828
2	1 to 2	2	2	48.01395	43.3484
3	1 to 3	3	3	4.662821	4.02705
4	1 to 4	4	4	0.635522	0.05761
5	1 to 5	5	5	0.577902	0.36892
6	1 to 6	6	6	0.208954	0.01164
7	1 to 7	7	6	0.197304	0.06455
8	1 to 8	8	8	0.132746	0.00509
9	1 to 9	9	9	0.127653	0.00012
10	1 to10	10	9	0.127526	0.00039
11	1 to 11	11	9	0.127133	0.00144
12	1 to 12	12	12	0.125686	0.00089
13	1 to 13	13	13	0.124794	0.00653
14	1 to 14	14	14	0.118259	0.00620
15	1 to 15	15	14	0.112054	0.03644
16	1 to 16	16	16	0.075605	0.01350
17	1 to 17	17	16	0.062099	0.03904
18	1 to 18	18	18	0.022913	0.00785
19	1 to 19	19	19	0.015346	0.02126
20	1 to 20	20	20	0	---------

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

From Table 1 the error curve has been obtained, Fig. 4. With the first found m<1.0, the scanning 4 has been selected, the curve shows that the error has a marked decrease until scanning 4, without significant changes from scanning 4 to end. 4th order polynomial for both X data and Y data correspond to the scanning 4 (M=4, N=4). The polynomials M=4 and N=4 show fit to the real shape of the exercise A. The curve obtained for exercise A versus demonstrated curves are observed in Fig. 5. In the case of the exercise A, 4th order polynomials are a good choice in terms of error, for both the X curve and the Y curve, also 4th order polynomials exhibit acceptable behavior to follow the real shape of the exercise A.

The same procedure has been applied to the other three rehabilitation exercises. In exercise B, the first m<1 has been found in scanning 4. The *eXYP* exhibits marked decrease until scanning #4 corresponding to 4th order polynomials for both X and Y data. These polynomials provide a curve acceptable regarding demonstrations, Fig. 6. In exercise C, the first m<1 has been found in scanning 3. The *eXYP* exhibits marked decrease until scanning #3 corresponding to 3rd order polynomials for both X and Y data, obtaining the curve (X, Y) observed in Fig. 7. Fourth order polynomials have been also selected for exercise D. The selection has been also made according to the decrease of the *eXYP* curve with the first m<1 in scanning 4, obtaining the curve of Fig. 8 for X and Y data.

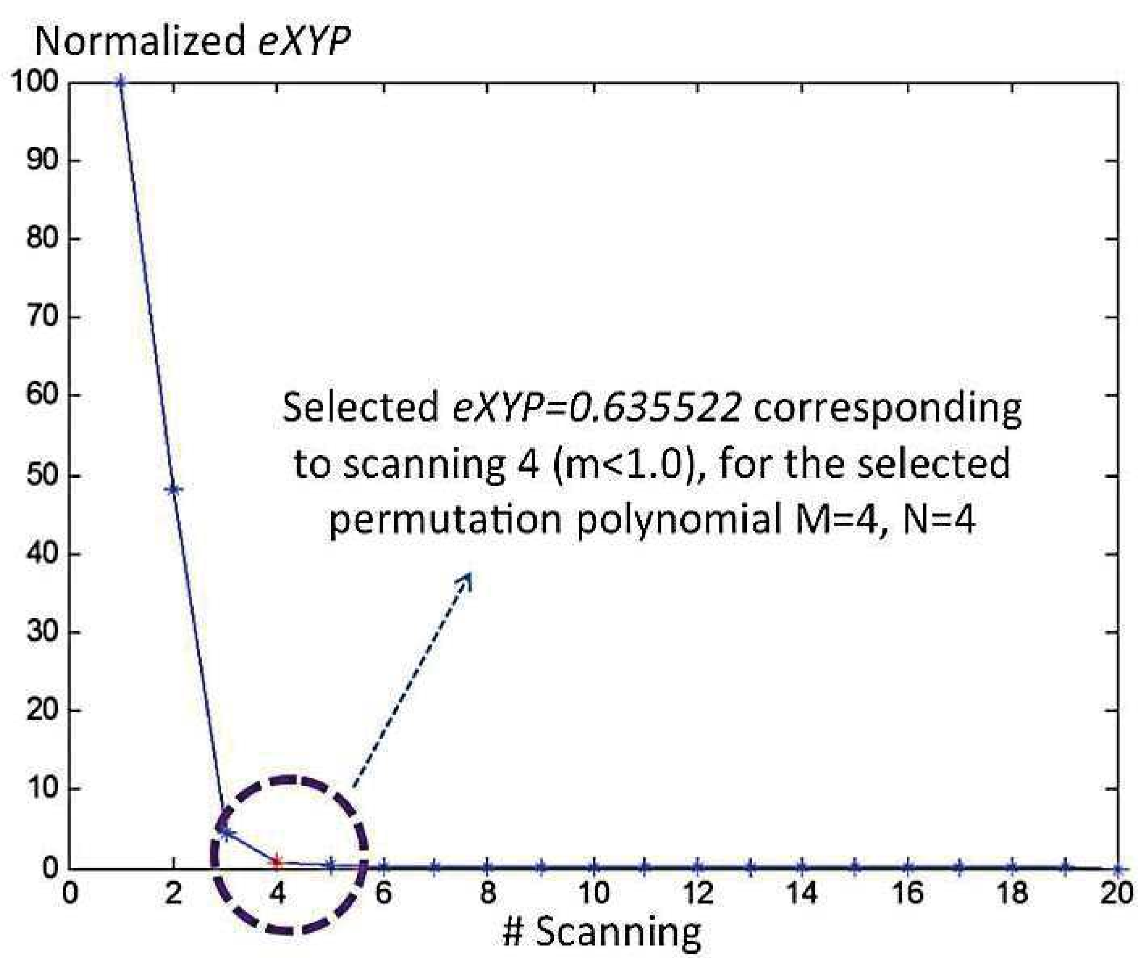


Fig. 4 *eXYP* curve - exercise A

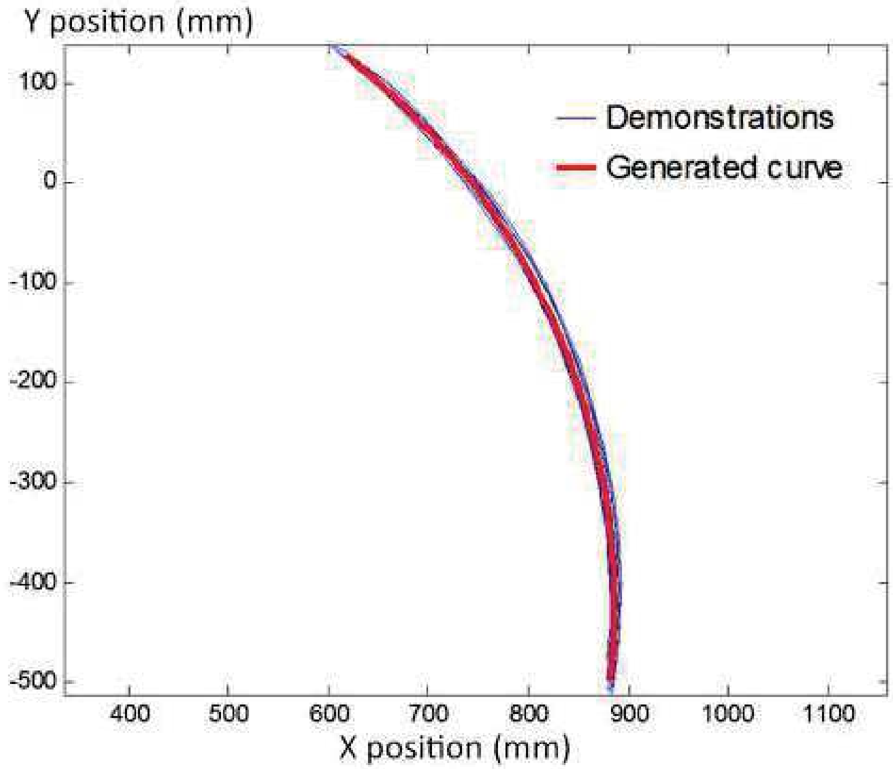


Fig. 5 Generated curve for the exercise A in the Y and X axis with 4th order polynomials

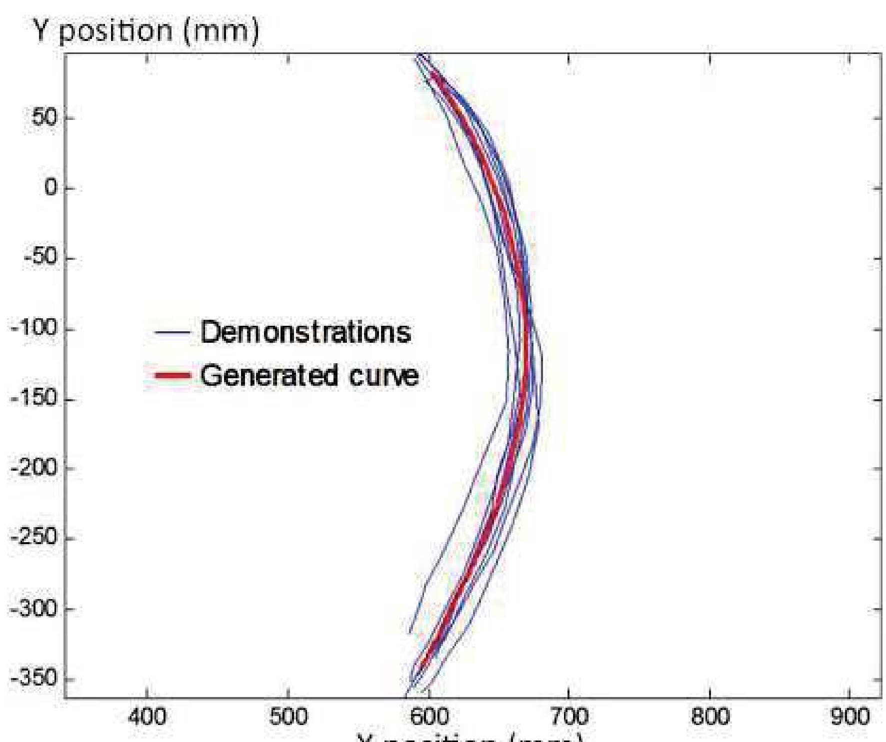


Fig. 6 Generated curve for the exercise B in the Y and X axis from 4th order polynomials

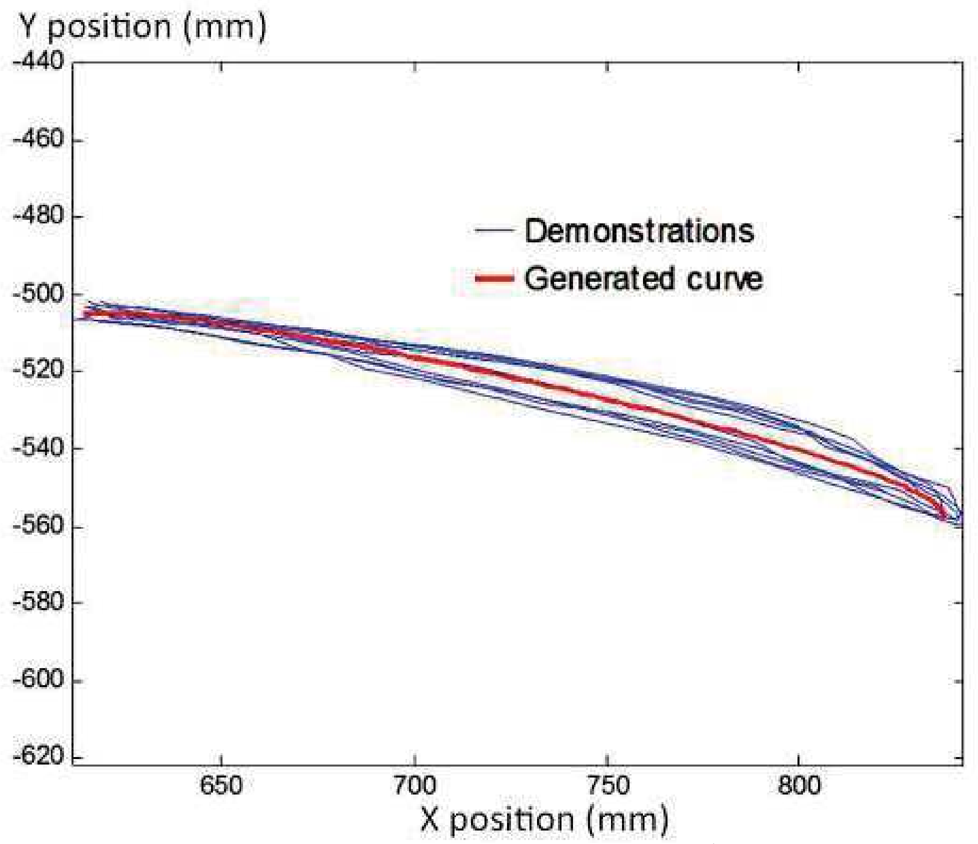


Fig. 7 Generated curve for the exercise C in the Y and X axis from 3rd order polynomials

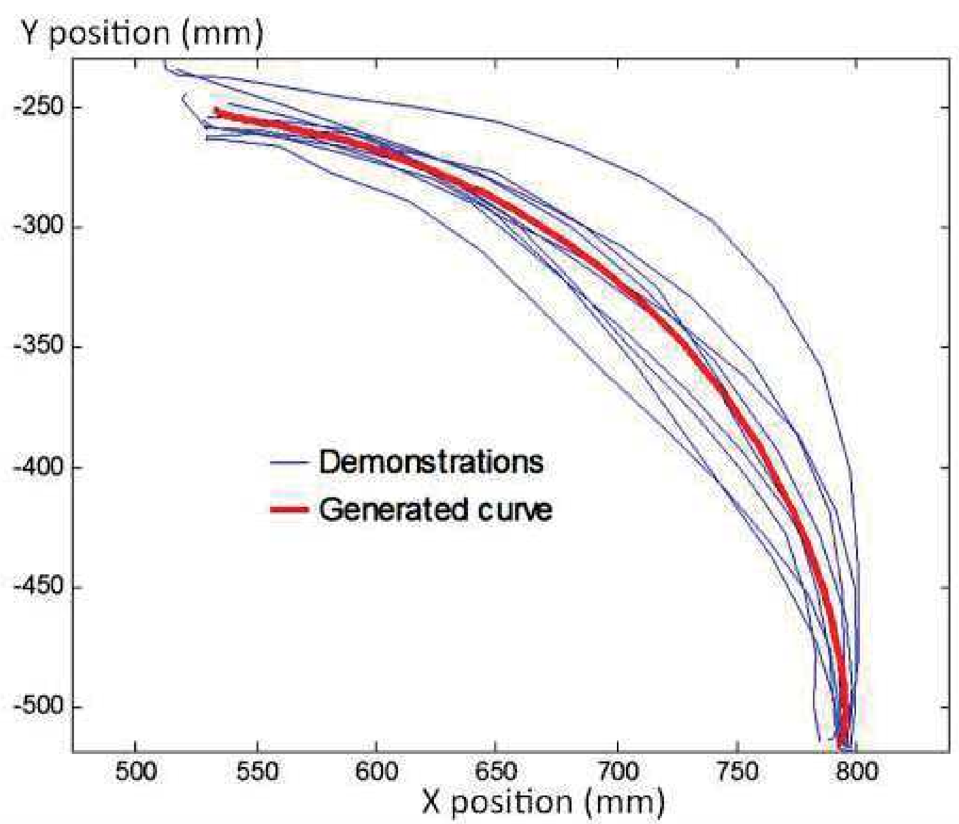


Fig. 8 Generated curve for the exercise D in the Y and X axis from 4th order polynomials

The generated curves fit successfully to the real shape of the knee exercises. Polynomials varied between the third and fourth order; fourth order for exercises A, B and D, which describe curves formed by the rotation of the hip or curves formed by the combined rotation of the knee and hip; third order polynomial were selected for exercise C which described trajectories less bend than the other exercises.

The device could reproduce the reference trajectory by using the average of the velocities and accelerations of all the stored trajectories or by using velocities and accelerations into a range selected by the therapist.

IV. SELECTING THE ONE OF THE DEMONSTRATED TRAJECTORIES

The generated reference curves by regression can be reproduced by the device since it represents the entire set of stored trajectories. However, in order to offer a second option to the therapist, the stored trajectory that best fit all the trajectories has been selected by using as reference points the curve generated by polynomials. Thus, the trajectory has been chosen by evaluating the same error eXY_d (Eq. 3) calculated in the section 3.

Figs. 9, 10, 11 and 13 show the selected demonstrations versus the generated curves (x,y) for exercises A, B, C and D. As it can be seen observed in the figures, the selected demonstrations for exercises A, B, C and D are nearest to the generated curve and represent successfully the shape of the knee exercises.

The knee assistive device could reproduce the selected demonstrations with the same velocities and accelerations or by using velocities and accelerations into the ranges selected by the therapist.

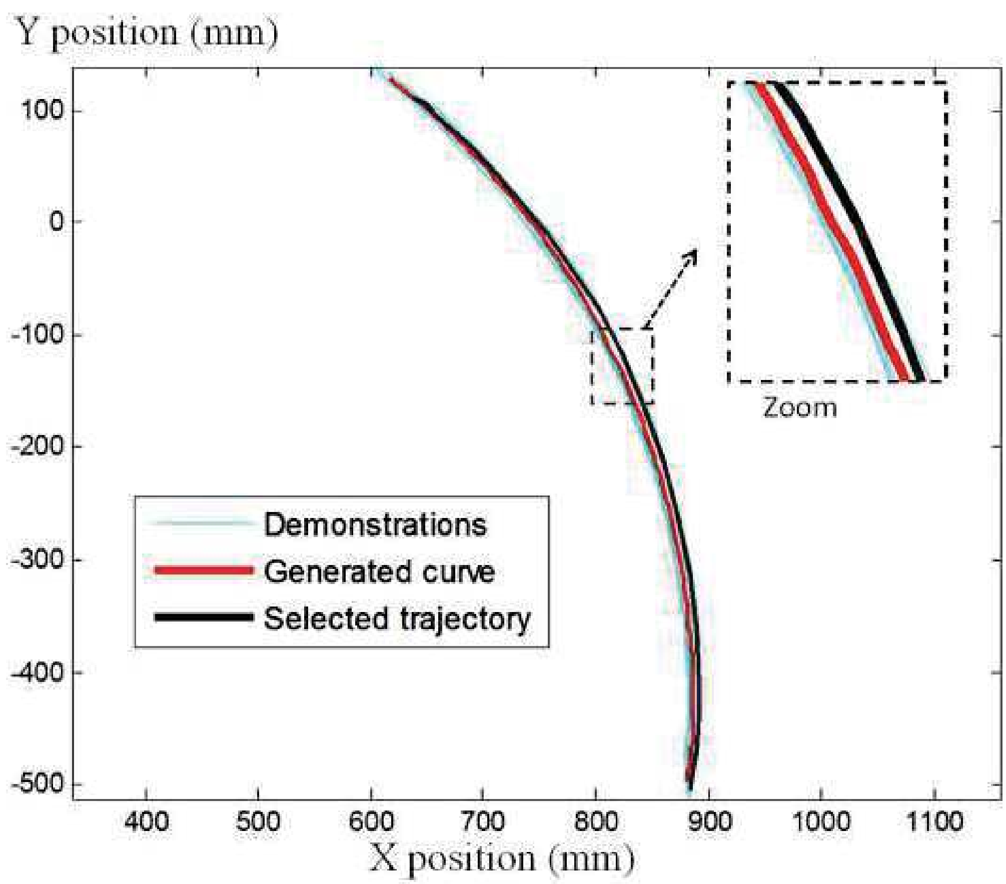


Fig. 9 Trajectory selected for X, Y axes, exercise A

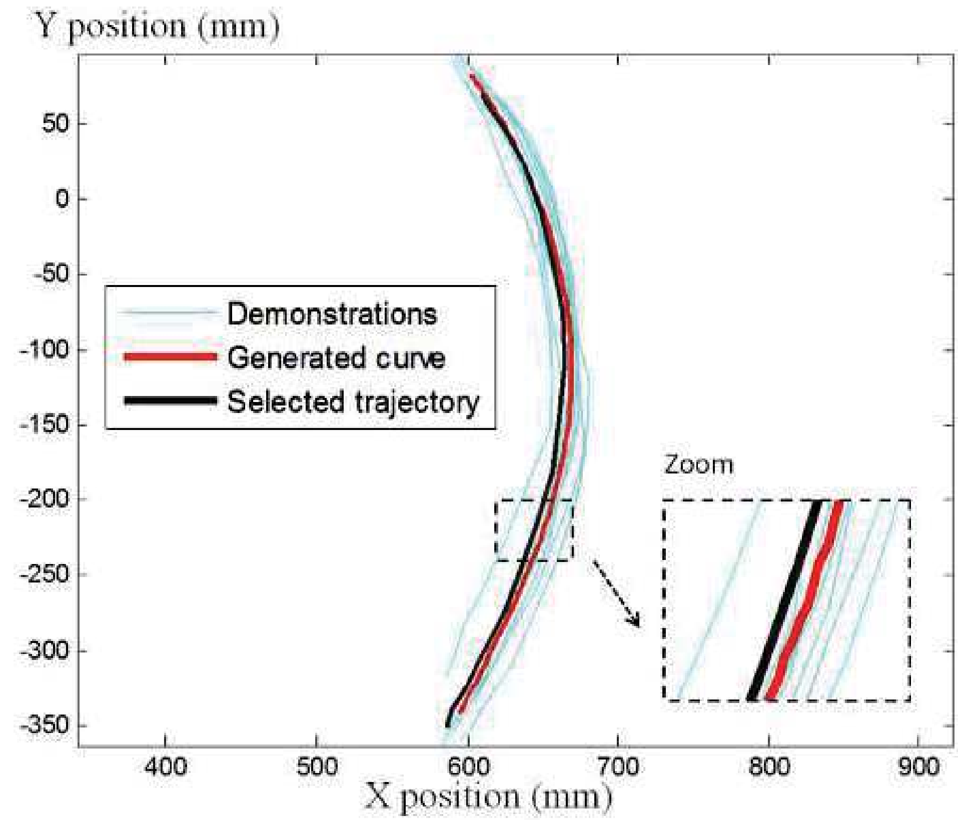


Fig. 10 Trajectory selected for X, Y axes, exercise B

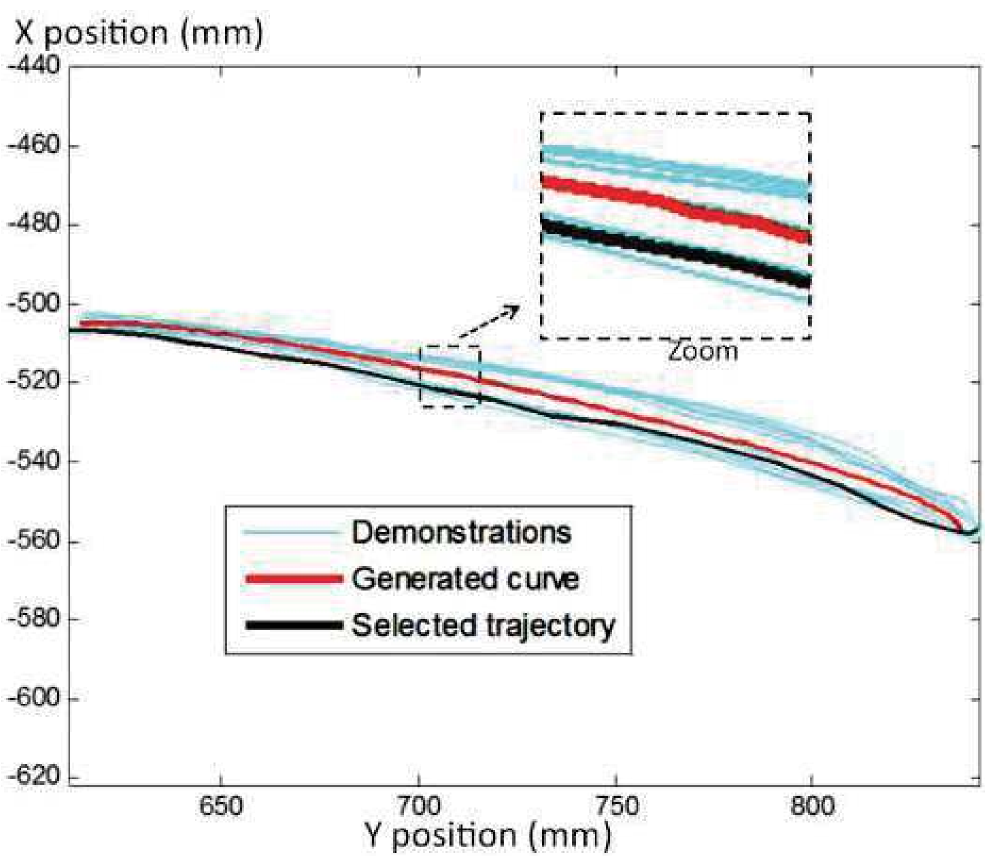


Fig. 11 Trajectory selected for X, Y axes, exercise C

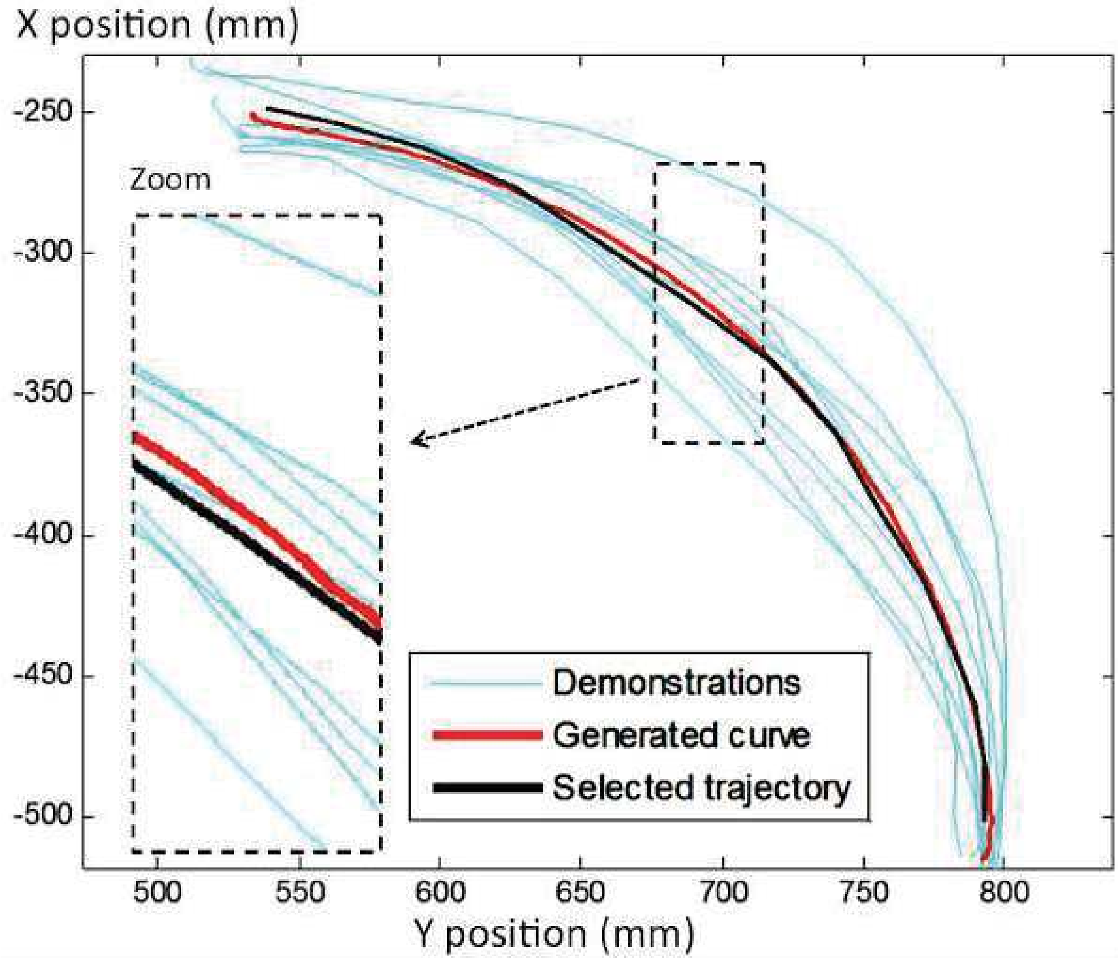


Fig. 12 Trajectory selected for X, Y axes, exercise D

The demonstration nearest to the curve generated by the polynomials represents the entire set of demonstrations; the selection provides a known trajectory for patients who can only execute one of their own trajectories due to the severity of their injury or illness.

V. CONCLUSIONS

In this paper a method to generate reference trajectories for a knee assistive device is proposed. The method successfully generates a reference trajectory, through regression analysis, by using a set of trajectories demonstrated on the assistive device.

The method is able to follow the shape of the proposed knee exercises, selecting effectively the order of the polynomials that provide a better fit to the real shape of the trajectories. The procedure to applied the method consists of four steps: 1) The patient performs the exercise on the assistive device while the therapist guides the movement and the device stores a set of trajectories; 2) a reference trajectory is generated by using the proposed method for curve generation; 3) One among all the demonstrated trajectories is selected by using the error behavior; 4) the assistive device reproduces the generated reference trajectory or the selected trajectory.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

The reference trajectories generated by the method can be reproduced by using velocities and acceleration from the demonstrations or by using commercial ranges. As future work the device could have an adaptive control to take into account the strength when reference trajectories are reproduced.

Acknowledgments Authors thank to CRIQ specialists for their help in the selection of exercises for knee rehabilitation.

REFERENCES

1. H. Hung-Jung and C. Tien-Chi, Motorized CPM/CAM physiotherapy device with sliding-mode Fuzzy Neural Network control loop. Elsevier: Computer Methods and programs in biomedicine 96, p 96-107, 2009.
2. J. Weiner. Device and method for knee joint rehabilitation, Patent No US 7,695,416 B2. United States Patent, 2010.
3. T. Nef and R. Riener. ARMin-Design of a Novel Arm Rehabilitation Robot. Proceedings of the 2005 IEEE, 9th International Conference on Rehabilitation Robotics,June 28-July 1, IL, USA, 2005.
4. E. Akdogan and M. Arif, The design and control of a therapeutic exercise robot for lower limb rehabilitaion :Physiotherabot. Elsevier. Mechatronics 21 (2011) 509–522, 2011.
5. A. Billard, S. Calinon, R. Dillmann, and S. Schaal, Robot programming by demonstration, in Handbook of Robotics. B. Siciliano and O.Khatib Eds. Secaucus, NJ: Springer-Verlag, pp. 1371–1394, 2008.
6. X. Wenkang, C. Bing and E. Rogers, Cascade based Iterative Learning Control of Robot-assisted Upper Extremity Stroke Rehabilitation. IEEE Conference on Decision and Control, Italy, December 10-13, 2013.
7. L. Renquan, L. Zhijun , S. Chun-Yi and X. Anke, Development and Learning Control of a Human Limb with a Rehabilitation Exoskeleton. IEEE Transactions on Industrial Electronics, Vol. 61, NO. 7, July 2014.
8. S. Calinon, F. Guenter and A. Billard, On learning, representing and generalizing a task in a humanoid robot. IEEE Trans. Syst.,Man, Cybern., Part B, vol. 37, no. 2, pp. 286–298, Apr. 2007.
9. S. P. Chatzis, D. Korkinof and Y. Demiris, A Quantum-Statistical Approach Toward Robot Learning by demonstration. IEEE Transactions on Robotics, Vol. 28, NO. 6. December 2012.
10. B. D. M. Chaparro-Rico and E. Castillo-Castañeda, Tesis: “Dispositivo para la Rehabilitación de la Rodilla Basado en un Mecanismo Paralelo”, Instituto Politécnico Nacional-CICATA-Qro. 2014.
11. C. M. Bishop. Pattern Recognition and Machine Learning, Springer, Singapure. 738 p, 2006.
12. S. Vijayakumar and S. Schaal, Locally Weighted Projection Regression: An On Algorithm for Incremental Real Time Learning in High Dimensional Spaces, Proc. International Conference on Machine Learning (ICML) pp. 288–293, 2000.
13. S. Calinon, F. D’halluin, E. Sauser, D. Caldwell, and A. Billard, Learning and reproduction of gestures by imitation: An approach based on hidden Markov model and Gaussian mixture regression, IEEE Robot, Autom. Mag., vol. 17, no. 2, pp. 44–54, Jun. 2010.
14. W. Yan, S. Yanyu and Y. Demiris, A morphable template framework for robot learning by demonstration: Integrating one-shot and incremental learning approaches. Elsevier:Roinversebotics and Autonomous Systems 62, p1517–1530, 2014.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 20, páginas 192 - 199.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Propuesta de un robot móvil para la detección de fuentes radioactivas

A. De la Barrera Gonzalez.[1], E. G. Hernández Martínez.[2], J. V. Cervantes Bazan.[3], J. A. Monterrubio Suarez.[4]

***Resumen*— En este reporte de investigación se presenta una propuesta para el diseño y desarrollo de un robot móvil terrestre, programado para realizar inspección, registro y una recolección de materiales radiactivos en el campo, ya que estas actividades suponen un alto riesgo para la salud humana; este robot móvil tiene tracción en las cuatro ruedas, es tele dirigido en una banda de frecuencia exclusiva de X-bee, cuenta con transmisión de vídeo en tiempo real en una frecuencia específica, se obtiene una ubicación del móvil con las coordenadas geográficas mediante GPS, así como se conoce la orientación geográfica del movimiento del robot mediante una brújula electrónica, contiene un reloj de tiempo real para conocer el momento en que se detecta la fuente radiactiva y por supuesto tiene un detector de rayos gamma; la información obtenida por cada módulo en el robot es recopilada por un microcontrolador AVR que se ha programado usando la plataforma Arduino y se transmite utilizando un módulo de X-Bee. El receptor desmodula la información y la entrega en formato digital utilizando otro módulo X-Bee, esta información también es administrado por un Arduino, se visualiza en una pantalla LCD y se almacena en una computadora personal a través de la comunicación USB, el vídeo recibido se presenta en una monitor de vídeo, la dirección de desplazamiento es manipulada por una palanca de mando que es operado por el Arduino.**

Palabras clave-- Robot, Arduino, Radiación, Geiger Muller, GPS, telemetría, teledirigido.

I. Introducción

El prototipo de robot móvil presentado en este reporte de investigación, tiene por finalidad inspeccionar el entorno, recolectar y enviar, información sobre actividad radioactiva mediante RF, con la cual se puede determinar si existe una fuente que produzca rayos gamma con la utilización de un contador Geiger Müller y con la finalidad de no exponer al ser humano e estas radiaciones debido a estas tareas. La telemetría es una de las tecnologías más importantes en la actualidad, cuando se trata de medir variables físicas, tales como; temperatura, humedad, tensión, intensidad eléctrica, ubicación geográfica, entre otras. Es importante conocer las condiciones del lugar donde se encuentre localizado el robot, para lo cual, se le implementó un sistema de visión con envío de señales al receptor de telemando; se le implementaron controles para dirigir el avance y dirección del robot enviados por radio control; contiene un módulo GPS para obtener la localización geográfica donde se ubique el robot y por lo tanto la probable ubicación de la fuente radioactiva; contiene una brújula electrónica para conocer la orientación y dirección de desplazamiento del robot. Con la finalidad de poder recolectar la fuente de radiación se le implementó un pequeño brazo robótico tele controlada para poder colocar o transportar la muestra a un contenedor apropiadamente protegido.

II. Antecedentes

Por muchos años el hombre, vestido y equipado de forma especial y por su propio pie ha detectado fuentes radioactivas cuando se sabe que existen, pero cuando no lo sabe se expone a las dañinas y posiblemente mortales radiaciones de fuentes radioactivas. Con la finalidad de no exponer al ser humano a estas fuentes se propone el desarrollo y uso de un robot móvil que posea el instrumental necesario para detectar y enviar una alarma o mensaje de advertencia al usuario, el robot que se propone es teledirigido para que le permita al usuario manipularlo para detectar fuentes radioactivas que puedan generar partículas alfa, beta y gama.

[1]Alfredo De la Barrera González. Docente e investigador del Tecnológico de Estudios Superiores de Ixtapaluca y del Instituto Politécnico Nacional, México. (e-mail: adelabarrera@ipn.mx).
[2]Eduardo Gamaliel Hernández Martínez Docente e investigador de la Universidad Ibero Americana Campus Ciudad de México (e-mail: eduardo.gamaliel@ibero.mx).
[3]Josué. V. Cervantes Bazán. Profesor Investigador de UAEM Texcoco y del Tecnológico de Estudios superiores de Ixtapaluca, México (e-mail: josue.vcb@gmail.com).
[4]Jesús Abel Monterrubio Suárez. Profesionista independiente (e-mail: ing.jesusabel@outlook.com).

Los robots móviles son mecanismos que contienen extremidades de cualquier tipo conocido, en este caso son ruedas de goma las cuales son capaces de producir fricción con el suelo, para así lograr desplazarse de un

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

lugar a otro siendo manipulados o programados por un usuario, para seguir una trayectoria programada de forma rutinaria o controlada por un usuario.

Los robots se han construido para sustituir al ser humano en tareas repetitivas dentro de algún sector laboral y en ambientes hostiles o riesgosos. En [1] y [2] se habla sobre un robot el cual es capaz de reorganizar varias cosas de entre un desordenamiento, se explora el problema de planificación de reordenamiento, donde un robot debe reorganizar varios objetos para que finalmente queden en la posición original se han utilizado modelos físicos simples, como cuasiestático inducido, para producir rápidamente planes intrínsecos eficientes de la ruta a seguir. Sin embargo, la complejidad de estos planes los hace particularmente sensibles a la incertidumbre del objeto a representar, los parámetros de la física, y la ejecución de una trayectoria. La robótica es una tecnología relativamente moderna en cuanto a su automatización mediante circuitos miniaturizados; la comprensión del funcionamiento y diseño de la robótica fija o móvil requiere del conocimiento de áreas de la ciencia y tecnología principalmente de ingeniería eléctrica, mecánica, electrónica de control y de sistemas computacionales, como se menciona en [3]. En [4], se habla de los efectos que tiene el robot cuando realiza tareas de manera autónoma y cuando las realiza de forma manipulada por un ser humano, pudiéndose entender el estudio independiente de estos movimientos o el estado colaborativo entre humanos y robots. Se puede hacer que un robot trace su camino de desplazamiento y determinar su ubicación geográfica mediante GPS, como se menciona en [5], en este trabajo, se explora el uso de objetos que pudieran colisionar. Mediante la utilización de láser para la ubicación de estos objetos se obtiene una ubicación precisa de los objetos y se traza una trayectoria de desplazamiento esquivando dichos objetos, como se menciona en [6]. Un sistema de telemetría consiste de uno o varios sensores o transductores de variables físicas que actúan como dispositivos de entrada, un medio de transmisión guiado o mediante radio frecuencia, procesamiento de estas señales, dispositivos remotos de grabación o visualización de los datos medidos

III. Desarrollo

En la figura 1A, se presenta el diagrama a bloques de los módulos de los que está compuesto el robot móvil, indicando cada uno de ellos. Como dispositivo de control y adquisición de datos se utilizó una tarjeta Arduino UNO, la cual mediante un programa recaba las variables desde el dispositivo GPS que proporciona las coordenadas geográficas de ubicación terrestre del robot; la brújula electrónica indica la dirección de desplazamiento; el RTC es un reloj de tiempo real que indica la hora y fecha real en la que se pueda detectar una fuente radioactiva la cual es detectada por un contador Geiger Müller, la tarjeta controladora de los cuatro motores para la tracción independiente de cada rueda, como medio de comunicación de radio frecuencia se utilizó un módulo X-Bee, en configuración half dúplex para dirigir la desplazamiento del robot móvil terrestre; la brújula electrónica indica la dirección de desplazamiento; el RTC es un reloj de tiempo real que indica la hora y fecha real en la que se pueda detectar una fuente radioactiva la cual es detectada por un contador Geiger Müller. La tarjeta controladora maneja los cuatro motores para la tracción independiente de cada rueda; como medio de comunicación entre el usuario y el robot móvil para el mando a distancia se utilizó radio frecuencia mediante un módulo X-Bee, en configuración half dúplex.

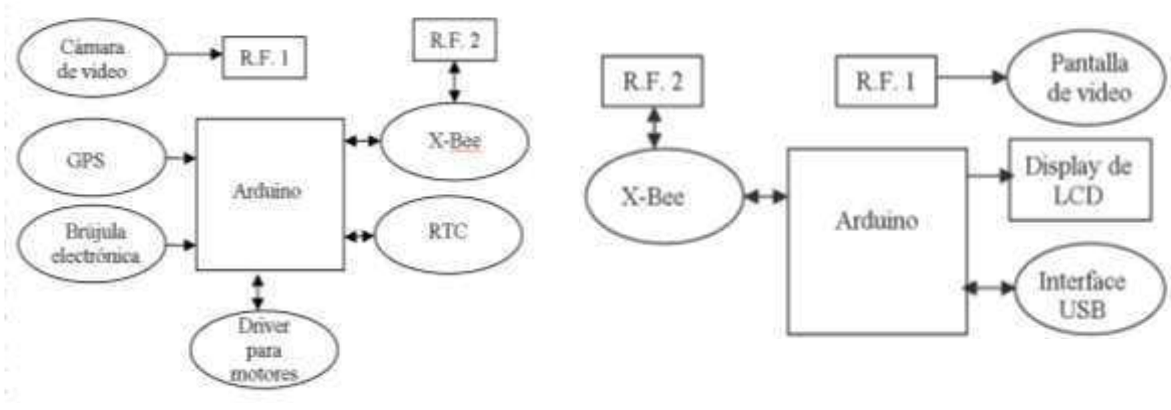
En la figura 1B, se presenta un diagrama de bloques del receptor, que está compuesto por un módulo X-Bee que entrega datos digitales a una tarjeta Arduino UNO para presentarlos en un display tipo LCD de 4 líneas y 20 caracteres y que, opcionalmente se pueden enviar a una PC mediante una interface USB, construida en el Arduino, también desde este módulo se envían al robot los comandos de desplazamiento utilizando un joystick y codificando el desplazamiento se envía al robot móvil mediante el radio modem X-Bee.

Para el diseño del robot móvil se utilizó una placa rectangular de PVC de 0.55x0.46m, donde se instalaron los cuatro motores, cada uno con una rueda de neopreno, para lograr una mayor fuerza de tracción y estabilidad en superficies planas y difíciles, estas ruedas están acopladas de manera que el vehículo pueda desplazarse en terrenos tales como asfalto, concreto, césped y algunas partes rugosas como terracería o escombros de construcción. El diseño rectangular de la plataforma, permite una mejor distribución del peso que lleva consigo, como son baterías, tarjetas controladoras de motores, sistema de visión FPV, blindaje de lámina de plomo de 500 gramos para circuiterías y la instrumentación de detección de radiación gamma. Cada una de los partes y módulos utilizados se describen a continuación:

1.- Sintra de PVC. Es una placa de PVC espumoso de 0.46x0.55m, resistente a altas temperaturas del ambiente, a golpes y a la humedad; es utilizada como soporte para toda la instrumentación electrónica, instrumentos de medición y en la que se montan los 4 motores que dan tracción al robot móvil.

2.- Arduino, como se menciona en [8] y [9] es una plataforma de desarrollo que incluye un microcontrolador como hardware y un lenguaje de programación; con ella se pueden realizar diversos proyectos sin necesidad de licencia para su uso, para el robot móvil se utilizó un módulo Arduino UNO, como se muestra en la

figura 2A, para la adquisición de datos digitales en el robot móvil y la recepción e interpretación de comandos de desplazamiento del móvil desde el comando del usuario. En la parte del receptor también se utilizó un Arduino UNO, para la recepción y presentación de los datos de telemetría en un display de LCD; se utilizó un Arduino MEGA, mostrado en la figura 2B, para el control de desplazamiento con joystick.



A) B)

Fig. 1 Diagramas a bloques A) del robot móvil y B) del telecontrol, receptor de telemetría y recepción de video.



A) B)

Fig. 2 Tarjetas de desarrollo A) Arduino UNO y B) Arduino MEGA.

3.- Motor con reductor EMG49. Es un motor de corriente directa (CD) el cual se muestra en la figura 3A, con una alimentación de 24 volts desarrolla una velocidad de 122 rpm, con un torque de 16 kg/cm, ya que contiene un sistema de engranes reductor de 49:1 [10]. En el interior del motor se encuentran acoplados dos sensores de efecto hall, los cuales conforman un encoder, estos sensores producen 588 pulsos por revolución del rotor, con la cuenta de estos puñsos se puede conocer con precisión el desplazamiento del móvil.

4.- Tarjeta controladora de motores MD49. Es una tarjeta controladora diseñada para manejar 2 motores de CD se alimenta con 24 volts, cuenta con selectores tipo jumper para seleccionar la velocidad de transmisión de datos ya sea a 9,600 o 38,400 bits/segundo, contando también con los conectores para las entradas de los encoders de los motores, la cual se muestra en la figura 3B, como se detalla en [11].

5.- Reloj en tiempo real. (RTC por sus siglas en inglés) que se presenta en la figura 4A, es un dispositivo electrónico el cuál proporciona la fecha y hora con precisión. Con este componente, el móvil será capaz de especificar la hora y fecha de la detección de la fuente radioactiva, como se indica en [12] y [13].

6.- Modulo GPS, *NEO-6 U-BLOX*. Es un dispositivo electrónico el cual a través de satélites geoestacionarios obtiene la ubicación en tiempo real del móvil, entregando como resultado parámetros tales como: latitud, longitud y altitud, este módulo electrónico se muestra en la figura 4A, y funciona como se indica en [14].

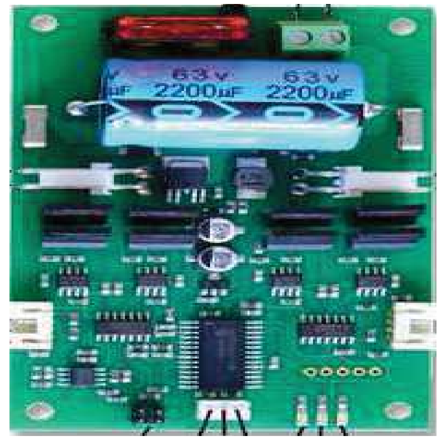
7.- Brújula magnética electrónica CMPS10. Es un dispositivo electrónico el cual cuenta con un magnetómetro que se alinea con el norte magnético terrestre. Este dispositivo se comunica mediante el protocolo I2C. Es

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

capaz de obtener la orientación que adquiere el robot en cualquier instante de tiempo, dicho componente se ilustra en la figura 5A, y funciona como se indica en [15].



A)



B)

Fig. 3 A) Motor con reductor EMG49. B) Tarjeta MD49, controladora de motores de C.D



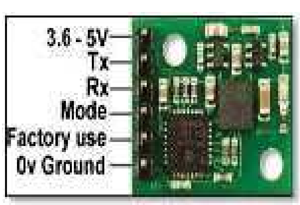
A)



B)

Figura 4. A) Tarjeta Tiny RTC, reloj de tiempo real, y B) Modulo GPS NEO 6 U-BLOX

8.- Joystick analógico. Es un dispositivo electrónico que contiene dos potenciómetros, uno para el eje de las abscisas y otro para el eje de las ordenadas; con este dispositivo se hace la manipulación del avance del robot para indicarle en qué dirección se debe de desplazar, se muestra en la figura 5B, como se detalla en [18] y [19].



A)



B)

Figura 5. A) Brújula magnética y B) Joystick analógico de dos ejes (X y Y).

9.- Detector Geiger Müller, Tipo GQ GMC-320 Plus. Es un instrumento de medición de rayos alpha, beta y gamma. El detector en su interior contiene un hilo conductor el cual actúa como el ánodo y la pared del dispositivo actúa como el cátodo. Además, en la parte interna del dispositivo contiene gas Argón, que al entrar en contacto con alguna partícula del haz de radiación ioniza el gas y desprende un electrón del Argón lo cual provoca un flujo de corriente que causa una avalancha de pulsos eléctricos detectables, como se muestra en la figura 6, se describe detalladamente en [16] y [17].

10.- Telemetría. Es una técnica de las comunicaciones que cuantifica y envía a larga distancia información sobre variables físicas. El dispositivo que se utiliza para la transmisión RF es el módulo de radio modem X-BEE PRO S3 de largo alcance, como se muestra en la figura 7A, Zigbee es un protocolo de comunicaciones inalámbrico basado en el estándar de comunicaciones para redes inalámbricas IEEE_802.15.4. Opera a una frecuencia de 2.4 GHZ con una velocidad inicial de transmisión de 3600 baudios/segundo. Cuenta con 8 canales digitales de transmisión de datos, como se detalla en [20]. El X-Bee convencional va montado a un adaptador como se muestra en la figura 7B. Estos radio módems pueden ser administrados por el programa XCTU, que fue desarrollado por la compañía Digi International, para configurar el modulo X-BEE se cuenta también con un monitor serial para enviar y recibir datos de algún otro modulo X-Bee con el que se encuentre enlazado.

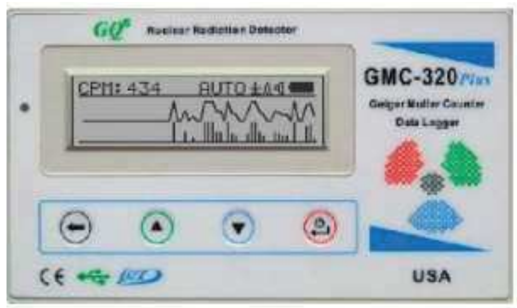
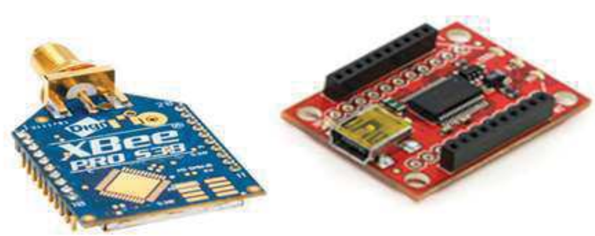


Figura 6. Contador Geiger Müller GQ GMC-320 Plus



A) B)

Figura 7. Módulo radio módem A) X-Bee PRO S3B, de largo alcance y B) base USB para radio modem X-Bee

11.- Sistema de Primera Visión de Vuelo (FPV), es una tecnología propia a los drones, pero se utiliza el mismo principio en el robot móvil, para que a larga distancia el usuario pueda observar las condiciones del lugar por las que el móvil realiza su desplazamiento de un lugar a otro. Los elementos que conforman un sistema FPV son: sensor de video como el mostrado en la figura 10, Transmisor, Antena, Receptor y Sistema de visualización.

La transmisión de audio y video con el sistema de visión FPV utiliza una frecuencia de transmisión de 5.8 GHz. La cámara utilizada se muestra en la figura 8, es de modelo CCTV-131 con un sensor de Semiconductor de Óxido de Metal Complementario (CMOS), con una resolución de 762 x 504 pixeles las imágenes se muestran con claridad, se utiliza la transmisión de video bajo el estándar NTSC (National Television Systems Committee, NTSC), y cuenta con un ángulo visual de 72°.

En la figura 9 se muestra el interior del gabinete del control remoto para desplazamiento del robot, donde se aprecia la tarjeta Arduino Mega, el módulo del Joystick, el módem X-Bee con su respectiva antena y una batería LIPO de 7.4 volts con capacidad de 1500 mAh.

Para la telemetría se instalaron dos módulos X-Bee de largo alcance, estos dispositivos son Half-duplex, ya que pueden funcionar tanto como transmisor o receptor pero en solo un momento dado y posteriormente intercambiar la función, de acuerdo al requerimiento programado.



Figura 8. Cámara CCTV-131

Básicamente existen tres formas de presentar los datos de la telemetría:

1. Mediante el monitor de datos del XCTU, utilizando una computadora tipo PC.
2. Mediante monitor serial de Arduino.
3. Mediante un display alfanumérico de LCD.



Figura 9. Componentes que conforman el control del robot móvil.

En el monitor serial del XCTU se observó que después de cierto tiempo de estar mostrando los datos de telemetría, este se suspendía y la pantalla no seguía actualizando los datos. Razón por la cual se procedió a realizar un programa en la tarjeta Arduino que mediante su monitor serial mostrará los datos de telemetría en una computadora PC, con lo que se corrigio la actualización de los datos.

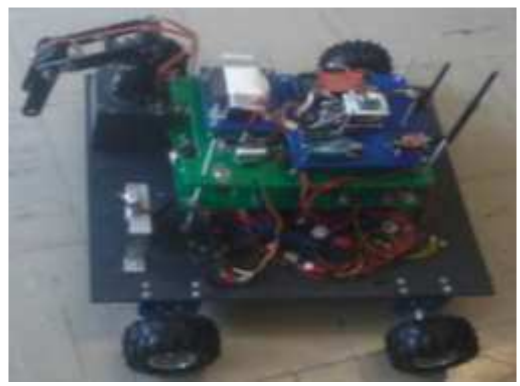
Para mostrar los datos de telemetría en una LCD, es necesario declarar otro puerto serial adicional en la placa Arduino con la librería SoftwareSerial. Con el uso de otra librería propia de Arduino llamada LiquidCrystal, con ella se pueden declarar los pines de Arduino que están conectados en los pines de display LCD 20x4, los pines de Arduino deben ser de salidas digitales. Por lo que con la utilización del dispaly de LCD es mucho más fácil recibir los datos de telemetría. La telemetría se envía mediante RF a 2.4 GHz a través de un módulo X-Bee. Para la parte de control e interpretación se utiliza un Arduino mega 2560, esta tarjeta cuenta con más pines tanto digitales como analógicos que los de la tarjeta Arduino Uno, como también cuenta con tres puertos seriales diseñados para la comunicación con otros dispositivos.

En la figura 10A, se muestra un segmento de código de programa para Arduino, la declaración de las diferentes librerías que se utilizarán para cada dispositivo utilizado, así como también, la declaración de variables de entrada, salida y variables para almacenamiento de datos. En la función setup(), de la figura 10, se inicializan las diferentes librerías que se declararon previamente. En el caso de la comunicación serial, se

inicializa el puerto serial a una velocidad de transmisión de datos de 9600 bits/segundo. El cuerpo del código se elaboró mediante funciones. Con este método de programación, en cualquier momento se pueden llamar desde una función principal, esto solo en la función loop(), pues lo que se encuentre programado en este apartado se repetirá de forma indefinida, es decir, es un ciclo repetitivo del programa, de acuerdo a las condiciones establecidas por el programador.

En la figura 10B se muestra el prototipo terminado el cual funciona de acuerdo a las características especificadas, el desplazamiento, la recepción de audio y video; así como las coordenadas geográficas y orientación del robot.





A) B)

Figura 10. A) Declaración de librerías de algunos dispositivos. B) Prototipo funcional terminado

IV. Pruebas y Resultados

Las pruebas del desplazamiento a control remoto del robot móvil se efectuaron en distintos sitios, se probó el funcionamiento en un laboratorio sobre piso de concreto, en dichas pruebas los comandos se ejecutaron apropiadamente.

Las pruebas de campo del sistema de visión (FPV) se realizan sobre asfalto. Esta prueba permitió controlar al robot móvil hasta una distancia de 400 metros. Con la cámara de video que se le instaló al robot móvil en la parte frontal, se recibió audio y video el cual con excelente claridad. Las pruebas de funcionamiento del módulo X-Bee que es transmisor en el robot se pudieron leer en el display de LCD, la ubicación, orientación,

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

tiempo real y contador de rayos gamma; y opcionalmente también se mostraron en una computadora Laptop Dell Inspirion 1420, utilizando el programa XCTU. La comunicación entre los radio módems X-BEE y la computadora se estableció satisfactoriamente.

V. AGRADECIMIENTOS

El presente proyecto robótico fue desarrollado para el Instituto Nacional de Investigaciones Nucleares (ININ), por lo que los autores agradecen el apoyo brindado por esta prestigiada institución, así como también agradecen al Tecnológico de Estudios Superiores de Ixtapaluca por el apoyo brindado para la realización de este trabajo.

VI. REFERENCIAS

[1] Takeo Kanade *The Vision and Autonomous Systems*, The Robotics Institute, Carnegie Mellon University. https://www.ri.cmu.edu/ 2015
[2] M.R. Dogar and S.S. Srinivasa., "Robust Trajectory Selection for Rearrangement Planning as a Multi-Armed Bandit Problem," The Robotics Institute, Carnegie Mellon University, 2015.
[3] M. W. Spong, S. Hutchinson, and M. Vidyasagar., R (2000) "Robot Modeling and Control," editorial JOHN WILEY & SONS, INC., New York.
[4] Anca D. Dragan, S. Bauman, J. Forlizzi, and S. S. Srinivasa., "Effects of Robot Motion on Human-Robot Collaboration," Carnegie Mellon University, 2015.
[5] M. Dogar, V. Hemrajani, D. Leeds, B. Kane, and S. S., "Proprioceptive Localization for Mobile Manipulators" Robotics Institute Carnegie Mellon University Pittsburgh, Pennsylvania 15213, february 2010.
[6] M. Dogar, V. Hemrajani, D. Leeds, B. Kane, and S. S., "Proprioceptive Localization for Mobile Manipulators," Robotics Institute Carnegie Mellon University Pittsburgh, Pennsylvania 15213, february 2010.
[7] Concepto de telemetría http://www.radiocomunicaciones.net/telemetria.html , consulta en línea el 03 de julio de 2015
[8] Concepto de Arduino, García González Antony, (2015) http://es.scribd.com/doc/168159411/Conceptos-basicos-de-micro-controladores-Conociendo-a-Arduino-pdf, consulta en línea, 07 de julio de 2015.
[9] Fabricante de Arduino, S/A http://mexico.newark.com/arduino, consulta en línea 07 de julio de 2015.
[10] Motor de 24V con encoder EMG49, http://www.superrobotica.com/S330105.htm, consulta en línea 25 de mayo de 2015.
[11] Tarjeta MD49, controladora de dos motores de C.D. http://www.robodacta.mx/index.php?subcats=Y&status=A&pshort=Y&pfull=Y&pname=Y&pkeywords=Y&search_performed=Y&q=md49&dispatch=products.search, consulta en línea 10 de junio de 2015.
[12] Hoja de datos del RTC DS1307, http://datasheets.maximintegrated.com/en/ds/DS1307.pdf , consulta en línea 02 de julio de 2015.
[13] Módulo de RTC https://www.dipmicro.com/store/DS1307BAT-MOD , consulta en línea 02 de julio de 2015.
[14] Módulo GPS NEO 6 U-BLOX, https://www.u-blox.com/en/product-search/field_product_category/position-time-152 consulta en línea 02 de julio de 2015.
[15] Brújula magnética http://www.robot-electronics.co.uk/htm/cmps10doc.htm, consulta en línea 02 de julio de 2015.
[16] Concepto de Geiger-Müller, http://fpaxp1.usc.es/~campus/doc/Geiger-Muller.pdf, consulta en línea 29 de abril 2015.
[17] Concepto de detector Geiger-Müller, http://bibliotecadigital.ilce.edu.mx/sites/ciencia/volumen2/ciencia3/094/htm/sec_8.htm, consulta en línea 29 de abril de 2015.
[18] Concepto de Joystick, http://www.glosarioit.com/#!Joystick, consulta en línea 04 de mayo de 2015.
[19] Hoja de datos del joystick , https://panamahitek.com/joysticks-en-arduino/, consulta en línea 04 de mayo de 2015.
[20] Módulo X-Bee, http://www.digi.com/lp/xbee/, consulta en línea 01 de junio de 2015.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 21, páginas 200 - 210.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Towards an adaptable human-machine interface for autonomous visual navigation for UAV in unstructured environments

Cesar Omar Orozco López[1], José Gabriel Ramírez Torres[2]

***Abstract*— In this work we present the first of two stages of development of a new Human-Machine Interface (HMI), adaptable to the context of current task for an Unmanned Aerial Vehicle (UAV) and operator's skills. This first stage consists on the implementation of a Three Layered Architecture (TLA) able to add, activate and deactivate reusable software components called skills or behaviors. The TLA and the skills are implemented on Robot Operating System (ROS) within the microprocessor onboard the UAV. The skills are added to the TLA using an abstract base class that allows to easily integrating the software components through a control mechanism. The skills, as well as their parameters and outputs must be registered in a skills library. The second stage, will involve the HMI design with an adaptive approach: its main objective will be balancing the information shown to the operator, according with the activated skills on the TLA as part of the UAV mission. The goal of this approach is the reduction of the human operators' workload, and therefore, the number of accidents caused by human factors, such as the operator workload and the ergonomic design of the HMI. This paper focuses on the TLA development, and it contributes with one of the first attempts of its implementation using ROS on a UAV quadcopter. The preliminary results of first stage of development with the workload measurements on novice operators using teleoperation-based HMI has been included.**

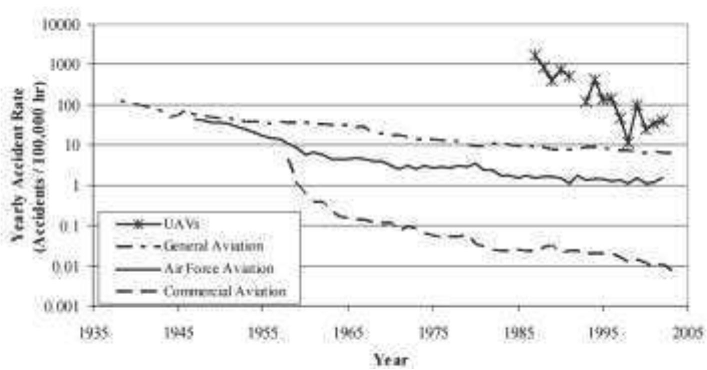
I. INTRODUCTION

Unmanned Aerial Vehicles (UAV) are defined as aircrafts without the onboard presence of pilots [1]. The UAVs are being used on a growing number of applications in the last decade, with potential usage in the civil UAV market in the next four years [2]. The new technological advances on electromechanical devices allowed the development of the UAV for commercial purpose with a reduced weight and size. These UAVs have been designed for different applications such as: search and rescue, fire combat, crop monitoring, border line monitoring, to mention a few.

The continued growth of use of these devices has produced a bigger interaction between the human and the UAV. The interaction has place if there is a Human-Machine Interface (HMI), which is used to exchange information between the operator and the UAV via a wireless channel. The operator sends navigation commands to modify the internal state of the UAV and the vehicle sends information related with the flight plan through the channel. The information received from the UAV is provided to the operator in the HMI. A Ground Control Station, the wireless channel, the operator, the UAV and the HMI are components of a bigger system known as Unmanned Aerial System (UAS).

The HMI has an important role at the UAS because is the only tool available for interaction between the operator and the UAV. The efforts to resolve issues related with the HMI have been scarce given how the research have been focused on the problems of automation and interoperability [3]. A critical aspect to keep in mind is the HMI design: if there is information overload or information underload, the HMI design could be the cause of risk situations for the UAS. Moreover, if the information displayed in the HMI is not clear the operator could be la cause of accidents, such those registered by the U.S. Army and Air Force from 1985 to 2002 [4]. The Figure 1 shows the number of accidents associated with an UAV compared with a manned aircraft.

Figure 1. Relation of accidents with differents classes of unmanned and manned aircrafts, image taken from [4].



Authors are affiliated with the Center for Research and Advanced Studies of the National Polytechnic Institute of Mexico, Carretera Cd Victoria – Soto La Marina Km 5.5, Cd. Victoria 87130 Mx (e-mail[1] : corozco@ tamps.cinvestav.mx, email[2]: grtorres@tamps.cinvestav.mx).

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

The U.S. Army has identified the main causal factors of accidents as: material factor, environmental factors and human factor [5].

- **Material Factor**: Includes equipment failure and damage that can result from design flaws, component or system failure.
- **Environmental factor**: Includes noise, illumination and weather conditions which can adversely affect the performance of the operator or the equipment.
- **Human factor**: Related with the human errors, which are mainly inherent to human design, function and behavior.

The human causal factor is the most cited at the accidents related with the unmanned systems. Some studies have described this causal factor between ranges from 60% to 80%, especially with the UAV Predator [6, 7]. The human causal factor includes six classes of human errors: operator workload, situation awareness, crew coordination, fatigue, ergonomic design and training.

The operator workload is defined as the combination of task demands or load factors, and an operator's response to those demands [8]. The operator is exposed to high levels of workload if at least two conditions are presents during the flight plan: 1) the operator is fully responsible of the flight without any automation processes and agents for decision taking; 2) the flight is taking place at a non-structured low altitude environment. An operator perceives a higher level of workload at the low altitude environments due to the dynamic nature of the environment. Ergonomics has established that the humans have limitations to process all the information generated by UAV flights. Although the rest of the causal factors also have importance, the operator workload and the ergonomic design are directly related with the operator and the HMI. Those factors impose changes not only on the flight plans at low altitude environments, but also they impose changes on the HMI design. An HMI adaptive approach is desired to minimize the accidents related with workload and increase the operator's efficiency. Breant A. Terwilliger *et al*. in their work "Advancement and Application of Unmanned Aerial System Human-Machine-Interface (HMI) Technology" [3] refer to the last two problems as "the lack of the optimized information" and the "lack of adaptability and flexibility" for effective HMI designs.

In this work we propose the development of a new adaptable HMI using the emergent mobile devices. This project is divided on two stages: the first one consists on the implementation of a Three Layer Architecture (TLA) implemented on ROS, for easy reuse of developed software components. The second stage, at present in progress, consists of the development of HMI with an adaptive approach to balance the information displayed to the operator of the skills activated by the sequencing layer on the TLA. This paper contributes with one of the first attempts of implementing a TLA on the UAV fields. The expectation for the second stage is to contribute with an effort on the HMI for UAVs, minimizing the human operators' workload during the UAV flight on non-structured environments reducing the accidents related with human factors and provide an adequate way of data presentation for making decision.

This paper is organized as follows. Section II provides concepts and background for this document. Also, it introduces the NASA-TLX instrument, which is used to calculate the operator's workload in this work. Section III introduces the proposed approach, the required infrastructure and the methods for the TLA implementation. Section IV presents the results of the execution of the TLA and validates the feasibility to implement the adaptive HMI. Next, the preliminary results of evaluating the operator workload using a traditional teleoperation-based HMI are included. Finally, Section VI presents the conclusions and the future directions for this work.

II. Background

This section describes basic concepts related with this research, such as: TLA, techniques for operator workload measurements and a description of ROS and the *mavros* package. The TLA was firstly introduced by Firsby [9] as a functional layout for reactive-deliberative robots. The basic design proposed by E. Gat is followed, which is based on the Firby's architecture. There are different methods to collect information about the operator workload. This section includes the different types of workload measurements and a list of criteria which allows choosing the correct method to get this measure. ROS is described in order to understand how its features make it a convenient tool to implement the TLA.

A. *Three layer architecture*

The TLA is widely used on mobile robots and is represented by three modules or components (see Figure 2). Its purpose is to organize the functionality of the robot according to the time required by the algorithms (skills or behaviors) to finish their tasks. The organization is realized according to whether the algorithms has an internal state, contains state reflecting memories about the past, or contains a state reflecting prediction about the future. E. Gat on his work "On Three Layer Architecture." refers to each module as layers and they are: 1) deliberation layer, 2) sequencer layer and 3) controller layer. The layers described by E. Gat are known also as planning layer, sequencing layer and skill layer [10].

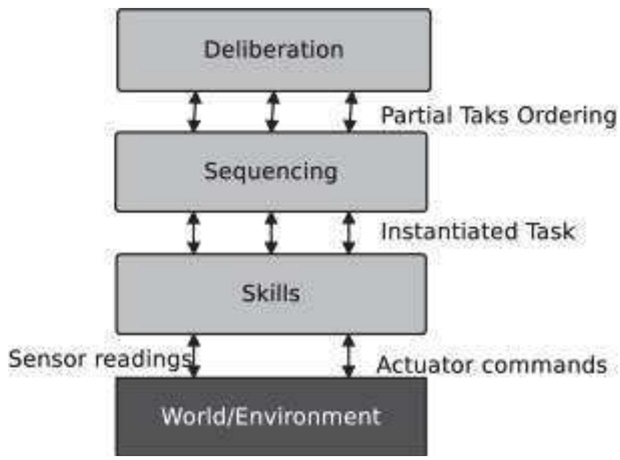
The planning layer is a container of algorithms that consume a long time to compute information to be used in the future. Usually the planning layer contains modules such as search-based and vision algorithms. It's usually considered as a deliberative planning capability that reasons in depth about goals, resource and timing constraints [11]. The algorithms allocated in this layer make a data projection to use the information in the future. The planning layer builds the plans to be sent to the sequencing layer.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

The sequencing layer is the one in charge to activate and deactivate skills while providing their parameters. An important feature of the sequencer is to not perform computations that take a long time, like any search or temporal projection. The activated skill modifies the work and the internal state to reach the targets established by the planning layer.

The skill layer is a dynamically reprogrammable set of reactive skills [11]. It consists of one or more computation threads that implement one or more feedback control loops [10]. This layer usually contains a library of functions called skills or primitive behaviors which are activated at any time by the sequencer or by an external input to this layer. Some examples of primitive behavior are: wall following, moving to a destination and follow target among others.

Figure 2. The Three Layer Architecure model proposed by E.Gat, based on Firby's architecture.



B. Robot operating system (ROS)

ROS, the open-source robot operating system is not an operating system in traditional sense of process management and scheduling. Instead of operating system, rather it is a framework that provides communication layers using a host operating system [12]. ROS is a framework to develop robotics applications widely used since 2007. Its features (peer-to-peer, multilingual, free and open-source, tool based and others) allow to developers write applications and run them in a short time using a set of ROS libraries or reusing code shared from others developers. The main aim of ROS is to allow the reuse of robotic software across the globe [13]. In fact, a piece of software made on ROS can be easily used in a different robot with small changes in its code [14, 15].

ROS supports different programming languages and even though C/C++, Lisp and Python are the main client libraries supported, a wide list of experimental clients are available [16]. The support of multiple programming languages allows developers to easily write code without the constraint of selecting a specific programming language.

The robotic applications are developed using a number of independent execution programs known as nodes. The nodes communicate with each other by publishing messages to topics (publish/subscribe messaging model). For example, a particular sensor driver could be implemented as a node, which publishes sensor data as a streaming message. The message can be sent using communication channels on ROS, called topics, and consumed by any other number of nodes, including higher-level systems.

ROS organizes the software in packages that can contain different components like: nodes, datasets, configuration files, or any useful module. A robotic application developed on this framework is a custom user package for specific purpose. A package might use packages available in the repositories, *e.g. mavros*, *mavros_extras* and *mavros_msgs* are packages that provide the message, topics and communication node for ROS with GCS and small unmanned vehicles. The *mavros* package uses the MAVLink protocol, a header-only message marshaling library for micro air vehicles. Despite ROS does not perform in real time it has been successful in the industry with the most popular platform using ROS: Aldebaran Nao [17]. ROS is also successful in the commercial market, and academic areas with at least 2724 mentions.

C. Workload measurement with NASA-TLX

The workload represents the cost of accomplishing mission requirements for the human operator [18]. If the operators have a high resource demand and the human cost of maintaining performance is unacceptable, the operator workload during the design and system operation must be evaluated. Miller provides three main classifications to measure the operator workload [19]:

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

- Performance measurements
- Physiological measurements
- Subjective measurements

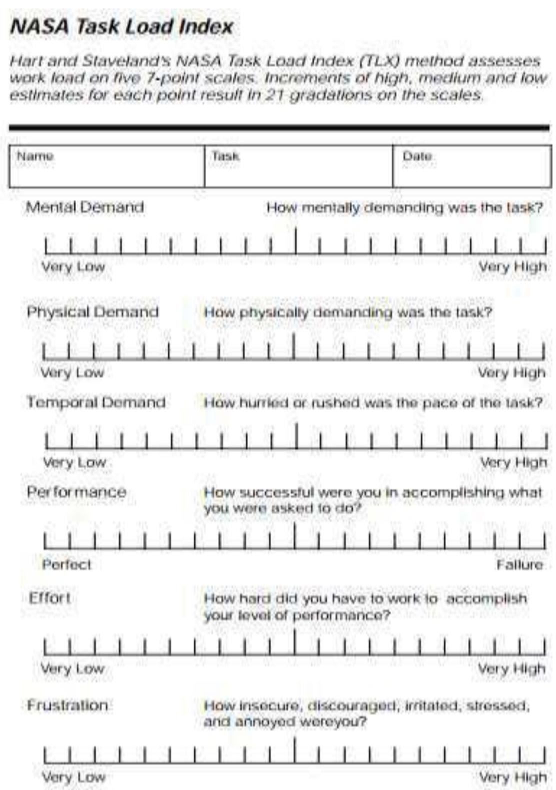
The performance-based methods provide two ways to measure the operator workload: 1) the primary task which it is based on the capability to perform a main task and, 2) the secondary task to measure the difference between mental capacity and total mental capacity. The physiological measurements use the human body reaction to get objective measurements. Usually require specialized equipment such as electroencephalograms and cardiac monitors. The subjective measurements are considered the simplest method to collect information of the operator workload. Usually this method uses questionnaires filled directly by the operator. This method is not accurate; however, it is widely used by its low cost and simplicity. In order to select a correct method to measure the operator workload, Kennet Boff et al. suggest consider the sensitivity, diagnosticity, intrusiveness, implementation requirements and the operator acceptance [20].

The NASA-Task Load Index (NASA-TLX or just TLX) is a subjective method to evaluate the operator workload (see Figure 3) [21]. It was developed by the Human Performance Group at NASA's Ames Research Center. This method consists of a questionnaire with six subscales that represent independent clusters of variables: Mental Demand (MD), Physical Demand (PD) and Temporal Demand (TD), Frustration (F), Effort (E) and Performance (P). The process to measure the workload using the TLX follows two steps: 1) familiarize the human operator with all the concepts of the NASA-TLX to be filled by the operator according the perception of the resource demand for specific task. Every subscale is pondered from 0 to 100 in 5-point steps. 2) The second step creates a weighting for every subscale through pairwise comparison. The TLX method has been used on areas of medicine, psychology and ergonomics to measure the workload using a HMI [22, 23, 24]. The NASA-Raw Task Load Index (NASA-RTLX) is a variation of the TLX method, created to reduce the time required to apply the evaluation. The NASA-RTLX eliminates the need of the pairwise comparison, simplifies the procedure and does not represent missing information on sensitivity [19]. The operator workload can be calculated using Equation 1:

$$OW = \frac{\sum_{k=1}^{n} S_k}{6} \quad \text{Eq. 1}$$

Where: OW is the Operator Workload, $1 \leq n \leq 6$, and S_k represents the k subscales of the NASA-TLX.

Figure 3. NASA-TLX questionnaire, the tool used to measure the subjective opearator workload, image taken from [25].



NASA Task Load Index

Hart and Staveland's NASA Task Load Index (TLX) method assesses work load on five 7-point scales. Increments of high, medium and low estimates for each point result in 21 gradations on the scales.

Name	Task	Date

Mental Demand — How mentally demanding was the task?
Very Low — Very High

Physical Demand — How physically demanding was the task?
Very Low — Very High

Temporal Demand — How hurried or rushed was the pace of the task?
Very Low — Very High

Performance — How successful were you in accomplishing what you were asked to do?
Perfect — Failure

Effort — How hard did you have to work to accomplish your level of performance?
Very Low — Very High

Frustration — How insecure, discouraged, irritated, stressed, and annoyed wereyou?
Very Low — Very High

E. Software In The Loop (SITL)

SITL is software that allows testing under simulation a Plane, a Copter or a Rover vehicle without any hardware. The simulator runs on a PC, to build the autopilot[1] code used for real vehicle prototypes and allows for testing of the behavior of the code. The simulator can run natively on Linux platform and other Operating Systems like Windows and Mac. The simulator works with the MAVLink protocol. The results of the SITL are identical to those produced when is executed using a real vehicle. Information about the MAVLink commands, simulation, testing using SITL and other relevant topics can be found at http://ardupilot.org/dev [18].

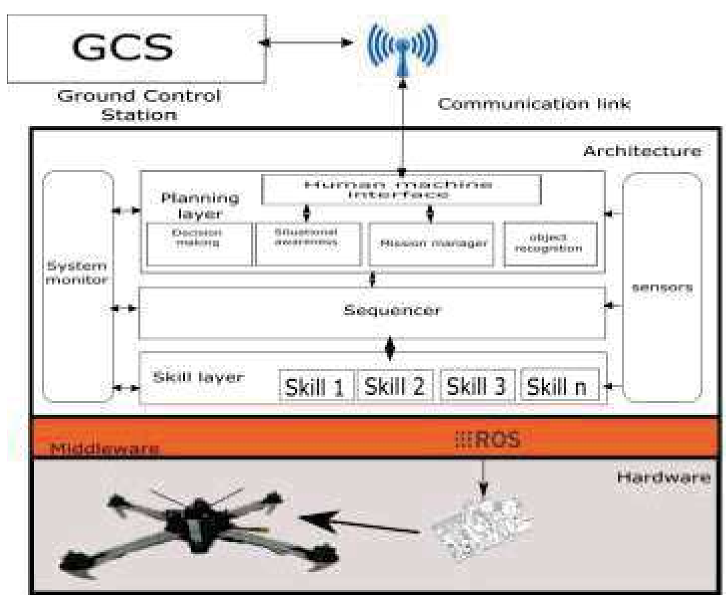
III. Methods

A. Proposed solution

A two-staged solution is proposed to achieve the final objective: to provide an adaptive HMI for reducing the operator workload and the accidents caused by human factors. The stages are: 1) the TLA implementation and 2) development of the adaptive HMI. Figure 3 shows the general schematic of the proposed solution. The architecture proposed is based on the TLA (described on the section II A) and widely used on mobile robots [26, 27]. The TLA is implemented on the microprocessor onboard of the UAV. The control commands and plannings, which are sent to the UAV by the operator using a wireless channel, are received in the highest layer of the TLA. The information received is evaluated to get the command type and evaluate its syntax. If the command type is a plan to be accomplished by the UAV, the planning layer provides the plan into sequencer. Once the sequencer layer receives a plan, it is processed to achieve the goals planned. The sequencing layer identifies skills declared on the plan and requests its activation to the lowest layer known as skills layer.

The lower layer includes a set of skills also known as primitive behaviors. The set of skills is compiled as a skills library and every single skill is a reusable software component. Additional information as parameters of the skills and the outputs are defined into the skills library. The outputs of the skills are reported through the HMI to the operator on runtime during a flight plan. If the active skill has achieved its specified goal, the "finished" state is reported to the sequencing layer and it sends a signal to the operating system to terminate the skill. The sequencer requests the activation of the next skill until the plan has been finished. The dynamic activation allows managing and balancing the information at the HMI to show just the data on runtime of the activated skill. The TLA has been implemented within the microprocessor on board of the UAV.

Figure 4. General scheme of the solution proposal. The three layer architectecture is implemented on a microprocessor onboard of the UAV.



B. Infrastructure

The mobile devices have been used to improve the interaction between the users and the interfaces. This work proposes the use of mobile devices due to their portability and immediate response to the choices selected by the user. A tablet (multitouch device) has been chosen to implement the adaptive HMI. A teleoperation-based HMI (traditional HMI) is also used to control and communicate with the UAV. Both HMIs will be used to measure the operator workload during a flight plan. The UAV utilized is a quadcopter assembled at the Center of Information Technology (Cinvestav LTI). The Figure 5 shows the UAV used in the experimentation. The details of the infrastructure are listed on the Table I.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Figure 5. Quadrotor UAV assembled on the CINVESTAV LTI using a microprocessor Raspberry PI andd NAVIO+.



TABLE I. DETAILS OF THE INFRASTRUCTURE

Device	Features
Multitouch device	Tablet Samsung sm-t560nu Quad core processor, 16GB internal memory, 9.6 in WXGA display.
Traditional HMI	Radio control Futaba T6EX model, 2.4GHz.
UAV	Raspberry Pi 2 model B, with 900 MHz, Quad core ARM Cortex-A7. 1GB RAM, OS Raspbian, telemetry modules, a controller NAVIO+ and rest of devices for the assembly.

C. First Stage: the three layer architecture

1) Planning layer

The planning layer is the highest level module. The planning layer receives the navigation commands and plans. It evaluates the syntax of command received using a set of regular expressions. The syntax rules were written using Flex and Bison, which are tools for building programs that handle structured input [16]. Flex is used to build lexical analyzers and Bison is often used to build parser generators. The TLA in the planning layer contains ROS nodes for exchanging commands and data flight between the operator and UAV, evaluating the type of the command received, establishing configurations and analyzing syntax of the plans. The set of regular expressions created to evaluate the syntax is small; however, it is easy to increase the capabilities of the TLA to extend the set of instructions to provide bigger flexibility. Some of the regular expressions used to evaluate the received commands are shown in the Listing 1:

Listing 1. Regular Expresions.

```
start: command_type asigna inst
command_type: cmd | pln | cnf
inst:
        PAR_I PARAL inst PAR_D inst |
        PAR_I behavior PAR_D inst |
        BEH_UPD PAR_I behavior PAR_D
        | JOYCMD_NAV PAR_I joycmd_nav PAR_D
```

The commands received on the TLA and evaluated with the syntax rules listed above are:

- **cmd**: execute commands that are not part of the skills library, e.g., enable the TLA for listen and execute a plan
- **pln**: execute one or more skills of the library organized to achieve a goal.
- **cnf**: execute configuration commands of the controller

2) Sequencing layer

The sequencing layer manages the plans to be executed. If the command type received in the planning layer is a plan and the syntax is approved, the plan is received by the sequencing layer and it is segmented into skills. A queue keeps the

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

activation order of the skills and the sequencer requests to the skills layer to confirm the existence of the skill. If the confirmation is received, the sequencer sends an activation signal to the *rospawn* node. Otherwise, the sequencing layer sends a message to the planning layer to abort the current plan.

3) Skills layer

The skills layer stores the skills that are activated by request of the sequencing layer. Each skill in the library allows the UAV to execute a specific task. A skill is a software component implemented in ROS as a node that can be activated dynamically, according with the plan provided by the operator and introduced using the HMI. The TLA includes the skills library in the lowest layer as a ROS node that reads an xml file to keep a registry of all the skills that the UAV is able to perform. The skills execute basic instructions such as takeoff, landing, or any other task that modifies the internal state of the UAV. The list below is part of the skills registered at the library:

- **arm:** Allows the UAV to listen navigation commands.
- **takeoff:** Request the UAV to reach certain altitude.
- **land:** The vehicle lands in the current position.
- **guided:** Change the operation mode of the UAV to guided mode that allows it dynamically guide the vehicle to a target location using telemetry radio module and GCS.

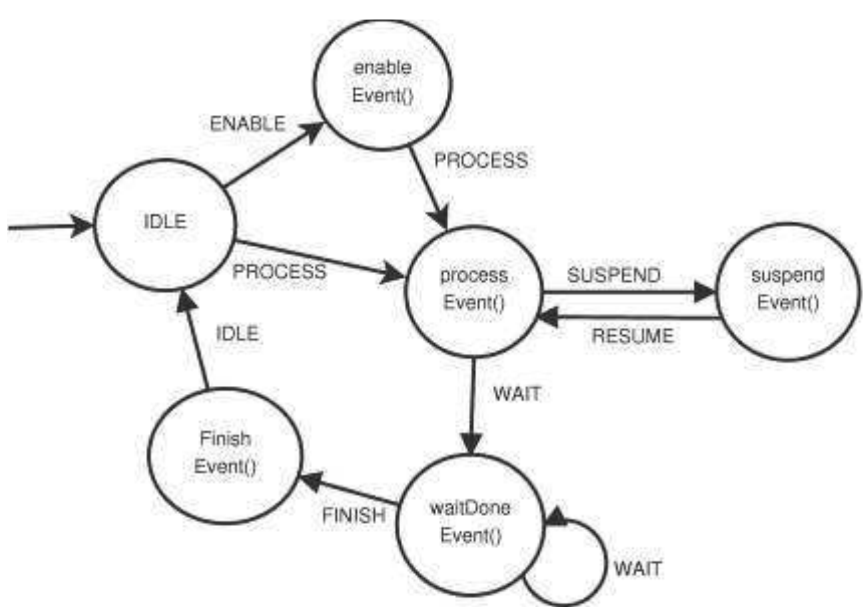
The skills are executed using the package *mavros* to control the actuators and sensors and modify the internal state of the UAV. If a reused component is added at the skill library, the component is modified to include an abstract base class called “behavior”. The class adds a finite state machine (FSM) that allows controlling the different states of the skill and reports the task progress. The FSM is shown on Figure 6 which describes the stages of a skill to reach its goal.

D. Second Stage: adaptive human-machine interface

The second stage involves the development of an adaptive HMI. The adaptive approach utilizes the TLA to balance the information displayed to the operator. The information of the data flight displayed corresponds with the skills activated during the plan execution. If the information is adjusted to the task of the UAV, the operator workload is reduced. To validate the reduction of the operator workload the subjective instrument NASA-TLX will be used to measure it (RTLX). The values of the operator workload will be compared with the values using the teleoperation-based HMI. The preliminary results of the measurement the operator workload using the teleoperation-based HMI are included in this paper.

A total of ten untrained volunteers (operators) participated at the evaluation of the operator workload using the teleoperation-based HMI. Participants were screened with two requirements: 1) at least high school level education and, 2) no motor or visual disabilities. Every operator was required to perform the activity called *task_1*. The task has three waypoints that must be reached by every operator. Once every waypoint is visited, the operator moves the UAV to the starting point known as HOME to finally land the UAV. The operators completed the *task_1* in the first scenario using the traditional HMI. At a second scenario, the operator must complete the *task_1* using the adaptive HMI. Each operator is requested to fill up the NASA-TLX questionnaire after finishing the task for both scenarios to evaluate the operator workload.

Figure 6. Finite state machine included in the abstract class to reuse software components.

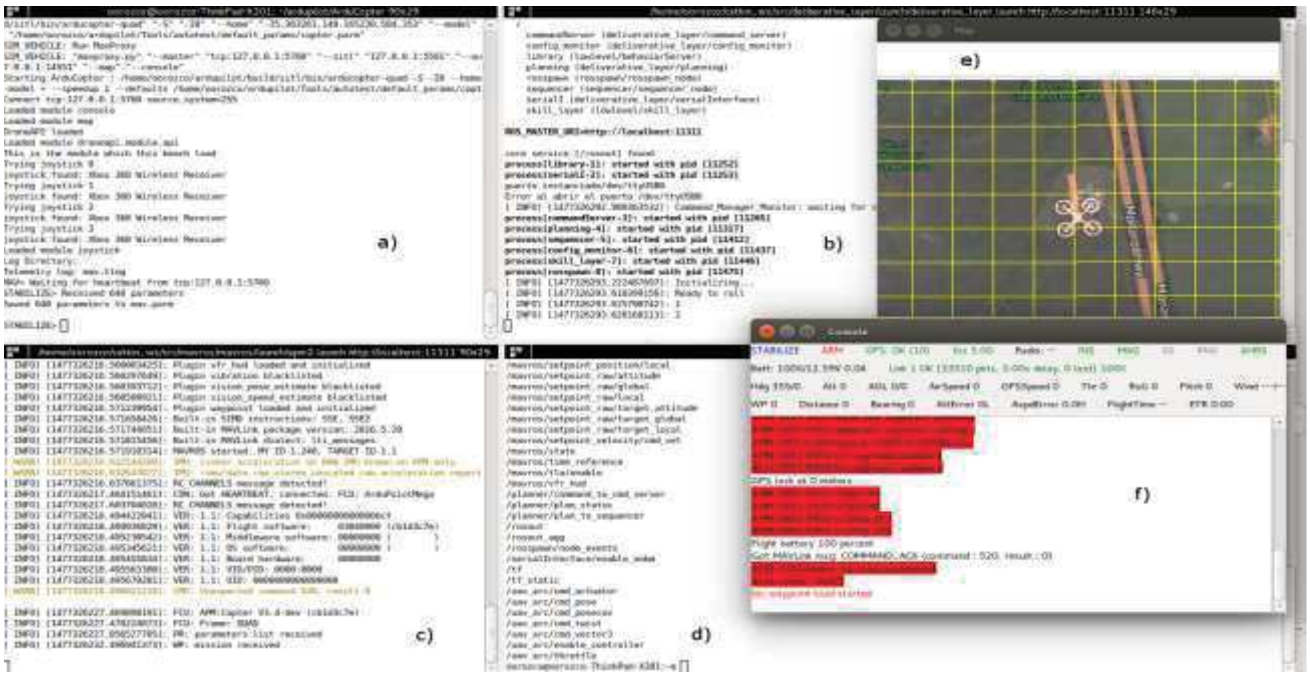


IV. RESULTS

A. Implementation of the three layer architecture

The TLA capabilities were tested using software in the loop (SITL) simulation and linking the *mavros* package with the Flight Control Unit (FCU). The Figure 7 shows the SITL and ROS running the *mavros* package and the TLA during the testing phase.

Figure 7. Output of the SITL linked with ROS. The console and the virtual map show the behaviors of the UAV during the execution of a plan. The terminal show the topics used by the TLA, the *mavros* node, and the *mavlink* protocol that communicates the virtual UAV and ROS. The graphic a) shows the SILT running, b) shows the TLA waiting for plans or commands, c) Mavros packages, d) List of the topics actually available, e) and f) the map and the console to monitoring the UAV behavior during the simulation.



SITL allows running a simulation of a quadcopter UAV directly on a computer, emulating the result of the execution of navigation commands as in the real environment. Mavros allows the use of the MAVLink protocol to extend the communication between ROS and GCS. The Figure 8 shows the TLA and its nodes in execution and the communication buses (topics) between the nodes before processing the plans received from the HMI. The commandServer node sends the plans for the planning node to analyze the syntax and approve it. During the idle phase, the TLA does not have any communication with the skill layer. To evaluate the TLA functionality a plan is sent to the UAV, the behavior of the UAV is then validated using SITL. The plan used to test the UAV is shown Listing 2.

Listing 2. A plan used to evaluate the UAV behavior, it request to the vehicle change its operation mode and changes it to guided mode, arm the vehicle and takeoff to 10 meters as target.

“pln <- (guidemode)(arm)(takeoff args:_altitude = 10)(land)”

The instruction sent to the UAV requests a change from the operation mode to guided mode, execute the arm command and reach an altitude target of ten meters for the takeoff command. Once the target is reached, the UAV goes back to land. The Figure 9 shows the communication buses between the guidedMode skill while a plan is processed. The *mavros* node reports the operation mode through the /mavros/state topic. Each skill in the queue of the sequencer is validated upon request a Boolean response from the skill library if the skill exists. If the skill reaches its goal, the “finish” status is reported to the sequencer and its deactivation is requested. The *rospawn* node keeps the list of the active skills using the process id (*pid*) assigned by ROS. The deactivation of the skill is processed terminating the process using its *pid*. The dynamic activation of the skills is used to adjust the information balancing on the adaptive HMI. To reduce the operator workload, only the output generated by the active skill is reported.

Figure 8. The TLA before the activation of the registered skills in the library. In the idle phase the sequencing layer does not keeps communication with the skill layer to modify the internal state of the UAV.

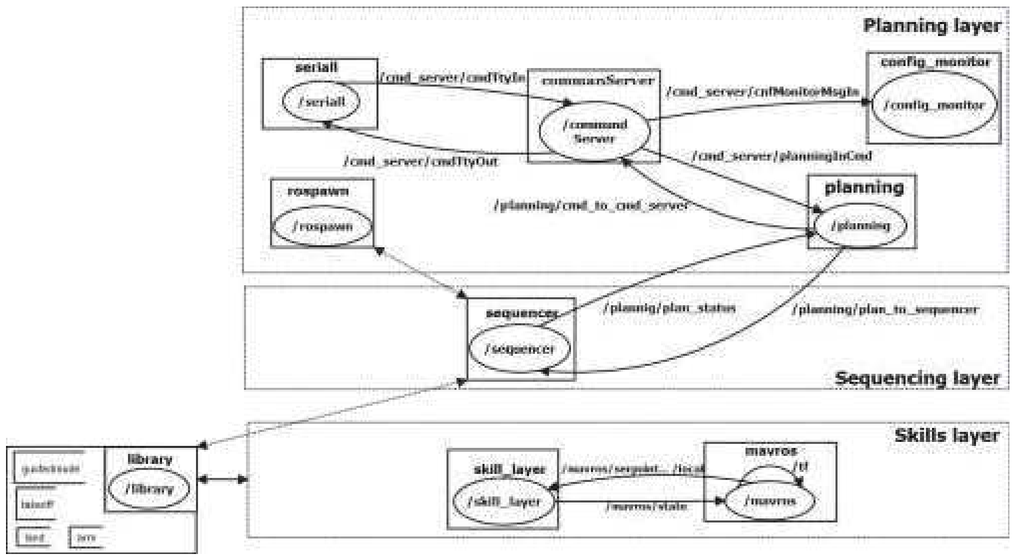
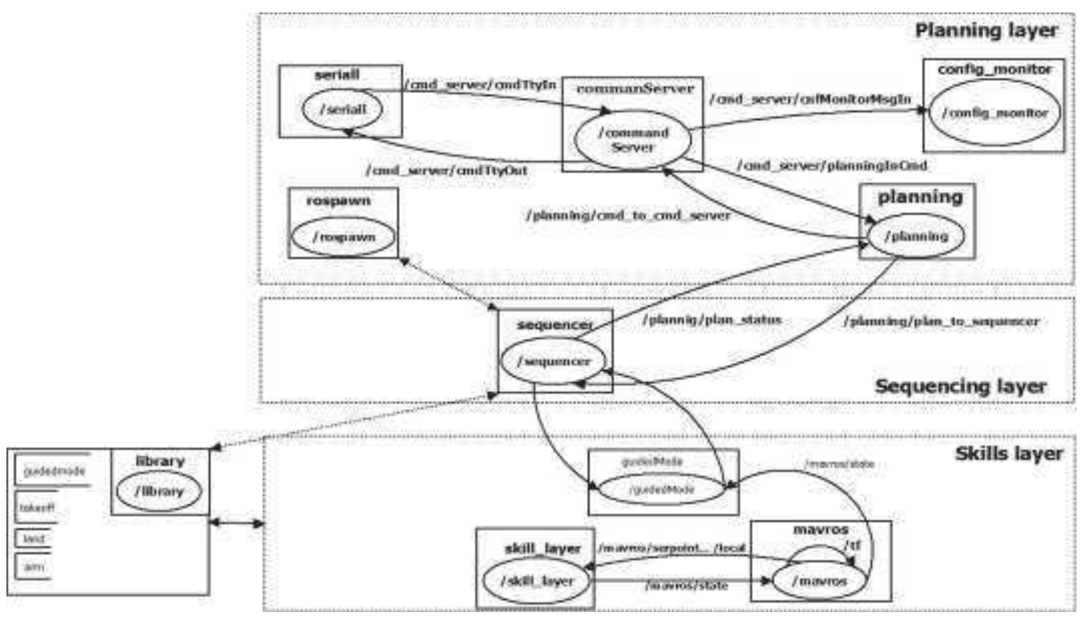


Figure 9. The TLA during the execution of a plan and the activation of skills. In the phase of execution, the sequencing layer request the confirmation if there is a skill registered on the library node called as the skills to activate. If a skill is active, it is able to get feedback from the mavros package about the current internal state, i.e. guidedMode.



B. Operator workload (preliminary results)

The operator workload was used as performance measurement. The Figure 10 shows the results of the total operator workload of the ten untrained operators that participated in the experiment. The measurements are represented just to the first scenario using the teleoperation-based HMI. The metrics were calculated using the method NASA-RTLX. The variable represented by the y axis is the total of the subscales evaluated (operator workload). The x axis represents the volunteers denoted by S_n where n = {1, 2… 10}. The operators were given a time limit of 15 minutes (due to battery time) to complete the task, without constraints in the number of attempts. Two of ten operators perceived a total demand of the task with values

lower that 50 percent. For the rest of them, the operator workload perceived is over 60 percent of demand during the task execution. The demand perceived for novice operators confirms the importance of reducing the demand, in order to minimize the risk of suffering a human caused accident. Table II shows the values of every subscale in the NASA-TLX of the operators for the experimentation in this paper.

Figure 10. Operator workload of the operators at the first scenario, the *x* axis represents the operator denoted by S_n, and the *y* axis represents the operator workload denoted as cost.

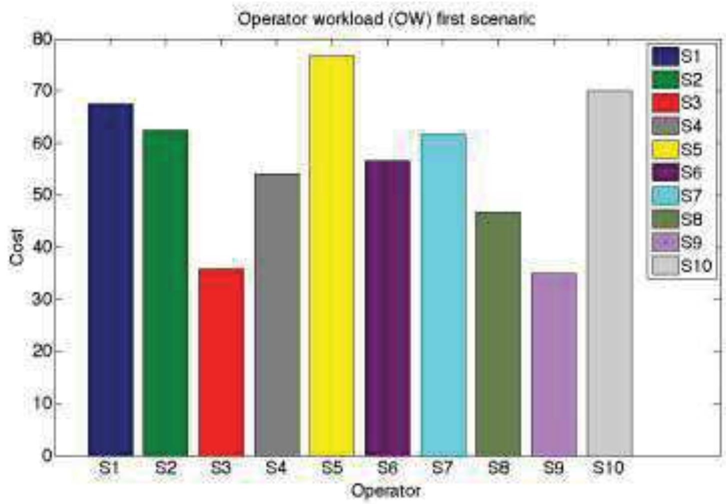


TABLE II. VALUES OF THE NASA-TLX SUBSCALE FOR EACH OPERATOR AT THE FIRST SCENARIO.

	Subjective value subscales						
Human operator	MD	PD	TD	F	E	P	**OW**
S_1	85	65	50	35	90	80	67.5
S_2	80	75	65	25	45	85	62.5
S_3	15	5	70	50	70	5	35.8
S_4	75	35	55	45	70	45	54.1
S_5	90	75	90	25	90	90	76.6
S_6	65	25	70	75	50	55	56.6
S_7	50	90	85	45	40	60	61.6
S_8	50	20	60	45	60	45	46.6
S_9	55	5	20	75	35	20	35.0
S_{10}	90	35	75	35	95	90	70.0

V. DISCUSSION

The use of the UAV on wide fields of application increases the interaction between the human and the UAV. If an operator has not enough expertise, the risk of an accident is bigger compared with expert users. A high percentage of UAV accidents are associated with human errors. The human causal factor includes the operator workload and the ergonomic design of the HMI. The preliminary evaluation of the workload presented in this work provides an idea of the high cognitive demand of novice operators to complete a small task and successfully control the UAV using a teleoperation-based HMI. This work presents an attempt to enhance the interface to balance the information with an adaptive approach and using multitouch mobile devices. The expectation is that balanced information presented on the display of the mobile device, allows the operator to be focused on the current context of the skill executed by the UAV, reducing the operator workload and the number of accidents. The dynamic reconfiguration feature of the TLA presented in this work is the base to achieve the reduction in the operator workload and balance the information in the adaptive HMI.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

VI. Conclusion

This work shows a general idea of a solution proposal; however, this paper is focused on the development and implementation of the three layer architecture and the results of its testing on simulated environments. This development contributes with one of the first attempts to implement this architecture on the field of the UAV, especially on a quadrotor with an onboard microprocessor, using ROS as main platform. The TLA and the inherent features by its implementation on ROS such as dynamic communication between nodes, allows to consider the development of adaptives human machine interfaces. This approach could be considered as an alternative to address the growing human-machine interaction in a trusty and safe manner.

References

[1] A. L. Salih, M. Moghavvemi, H. A. F. Mohamed, and K. S. Gaeid, “Modelling and pid controller design for a quadrotor unmanned air vehicle,” *in Automation Quality and Testing Robotics (AQTR), 2010 IEEE International Conference on*, vol. 1, May 2010, pp. 1-5.

[2] Frost and Sullivan: Study Analysing the Current Activities in the Field of UAV. European Commision, ENTR/2007/065(2007)

[3] Terwilliger, B. A., Ison, D. C., Vincenzi, D. A., & Liu, D. (2014, June). Advancement and application of unmanned aerial system Human-Machine-Interface (HMI) technology. In *International Conference on Human Interface and the Management of Information* (pp. 273-283). Springer International Publishing.

[4] R. E Weibel and R. J. Hansman, “Safety considerations for operation of different classes of uavs in the nas,” in *AIAA 4th Aviation Technology, Integration and Operations Forum, AIAA3er Unmanned Unlimited Technical conference, Workshop and Exhibit,* 2004.

[5] S. D. Manning, C. E Rash, P. A. LeDuc, R. K. Noback, and J. McKeon, “The role of human causal factors in us army unmanned aerial vehicle accidents,” DTIC Document, Tech. Rep., 2004.

[6] K. W. Williams, “A summary of unmanned aircraft accident/incident data: Human factors implications,” DTIC Document, Tech. Rep, 2004.

[7] Damilano, L., Guglieri, G., Quagliotti, F., & Sale, I. (2012). FMS for unmanned aerial systems: HMI issues and new interface solutions. *Journal of Intelligent & Robotic Systems*, *65*(1-4), 27-42.

[8] M. Mouloua, R. Gilson, J. Kring, and P. Hancock, “Workload, situation awareness, and teaming issues for uav/ucav operations,” in Proceedings of the Human Factors and Ergonomics Society Annual Meeting, vol. 45, no. 2. SAGE Publications, 2001, pp. 162-165.

[9] J. R Firby, “Adaptive execution in complex dynamic worlds,” Ph. D. dissertation, Citeseer, 1990.

[10] E. Gat et al., “On three-layer architectures,” Artificial intelligence and mobile robots, vol. 195, p. 210, 1998.

[11] R. P. Bonasso and D. Kortenkamp, “Using a layered control architecture to alleviate planning with incomplete information,” in *Proceedings of the AAAI Spring Symposium\ Planning with Incomplete Information for Robot Problems*, 1996, pp. 1-4.

[12] ROS: an open-source Robot Operating System In ICRA Workshop on Open Source Software (2009) by Morgan Quigley, Ken Conley, Brian Gerkey, et al.

[13] L. Josepth*, Learning Robotics Using Python*, P. P. Ltd, Ed. Pack, 2015.

[14] E. F. Aaron Martinez, Learning ROS for Robotics Programming, P. P. Ltd, Ed. Packt Publishing Ltd, 2013.

[15] B. G. Morgan Quigley and W. D. smart, programming Robots with ROS A PRACTICAL INTRODUCTION TO THE ROBOT OPERATING SYSTEM, I. OReilly Media, ed. Oreilly, 2016.

[16] [Online] http://wiki.ros.org/Client%20Libraries [Last access] 16- 10-22

[17] S. cousins, B. Gerkey, K. Conley, and W. Garage “Sharing software with ros [ros topic].” Robotics Automaiton Magazine, IEEE, vol. 17, no. 2, pp. 12-14, 2010.

[18] [Online]http://ardupilot.org/dev/index.html [Last access] 16-10-22

[19] S. Miller, “Literature review workload measures,” in University of Iowa.

[20] K. R. Boff, L. Kaufman, and J. P. Thomas, od perception and human performance. volume 2. cognitive process and performance,” DTIC Document, Tech. Rep., 1994.

[21] S. G. Hart, “Nasa-task load index (nasa-tlx); 20 years later,” in Proceedings of the human factors and ergonomics society annual meeting, vol. 50, no. 9. Sage Publications, 2006, pp. 904-908.

[22] C. M. Humphrey, C. Henk, G. Sewell, B. W.Williams, and J. A. Adams, “Assesing the scalability of a multiple robot interface,” in Human-Robot Interaction (HRI), 2007 2nd ACM/IEEE International Conference on, March 2007, pp. 239-246.

[23] Hohnson, C. A., Adams, J.A., & Kawamura, K (2003, October). Evaluation of an enhanced human-robot interface. In *Systems, Man and Cybernetics, 2003. IEEE International conference on (Vol. 1, pp. 900-905). IEEE.*

[24] Gunn, D.V., Warm, J. S., Nelson, W.T., Bolia, R. S., Schumsky, D. A., & Corcoran, K. J. (2005). Target acquisition with UAVs: vigilance displays and advanced cuing interfaces. *Human Factors: The Journal of the Human Factors and Ergonomics Society, 47(3),* 488-497.

[25] [Online]https://humansystems.arc.nasa.gov/groups/tlx/tlxpaperpencil.php [Last access] 16-10 20

[26] Schipani, S.P. (2003). An evaluation of operator workload, during partially autonomous vehicle operations. ARMY RESEARCH LAB ABERDEEN PROVIDING GROUND MD.

[27] Montemerlo, M., Thrun, S., Dahlkamp, H., Stavens, D., & Strohband, S. (2006, July). Winning the DARPA Grand Challenge with an AI robot. In *Procedings of the national conference on artificial intelligence* (Vol. 21, No. 1, p. 982). Menlo Park, CA; MA; London AAAI Press; MIT Press; 1999.

[28] Tucat, M., Gottifredi, S., Vidaurreta, F., García, A. J., & Simari, G. R. (2008). A layered architecture using schematic plans for controlling mobile robots. In *XIV Congreso Argentino de Ciencias de la Computación*.

[29] J Levine, *flex & bison: Text Processing Tools. O’reilly Media, 2009.*

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 22, páginas 211 - 220.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Dispositivo robótico coadyuvante para la rehabilitación de dedos de la mano

A. Zapatero Gutiérrez[1], J. F. Rodríguez León[1], J. F. Aguilar Pereyra[2], E. Castillo Castañeda[1]

Resumen- Se presenta el desarrollo y validación de un dispositivo robótico coadyuvante para la rehabilitación de dedos de la mano, el cual tiene por objetivo realizar movimientos asistidos de flexo-extensión de los dedos. El presente documento aborda desde el diseño del dispositivo, su correspondiente análisis cinemático y la validación del mismo a través de técnicas de procesamiento de imágenes. En base a los resultados, se obtiene un dispositivo reconfigurable donde las trayectorias correspondientes a los movimientos de flexión–extensión y extensión–flexión son semejantes y corresponden a los movimientos de las puntas de los dedos, corroborando de esta manera la funcionalidad del prototipo.

Palabras Clave: Rehabilitación Robótica, Flexo-extensión de los dedos de la mano

I. INTRODUCCIÓN

Las extremidades superiores son vitales en las actividades cotidianas de los seres humanos, en ocasiones sufren lesiones causadas por accidentes y/o enfermedades que disminuyen o impiden su funcionalidad. Los dedos de la mano son parte primordial y estratégica de estas extremidades, para enfatizar su importancia cabe mencionar que tan sólo al dedo pulgar se atribuye el 40% de la funcionalidad de la mano [1].

Los dedos de la mano son los encargados de realizar funciones como sostener un objeto, realizar trabajos de precisión y fuerza, detectar texturas, e inclusive sirven como extensiones del cuerpo para registrar sensaciones ambientales como el frío o el calor.

Actualmente uno de los métodos más usados para la rehabilitación de dedos son las llamadas férulas u órtesis dinámicas, que son dispositivos sanitarios que actúan sobre las articulaciones interfalángicas, aplicando presiones mediante diferentes apoyos y/o controlando los procesos de flexo-extensión de los dedos [2].

El problema de estos dispositivos es que suelen ser semielaborados y en algunos casos se confeccionan “a la medida” [2], lo que resulta en un método no sistemático. Eso sin contar que se requieren distintos tipos de férulas según el proceso de rehabilitación.

El impacto principal del dispositivo presentado en las siguientes páginas radica en el aspecto tecnológico y económico, ya que, al ser un dispositivo pensado para beneficio del sector de salud pública, permitirá optimizar los recursos al servicio de la sociedad.

Las terapias clásicas de rehabilitación son aplicadas por terapeutas especializados quienes se apoyan en sistemas de terapias manuales. Se ha comprobado que el uso de sistemas robóticos en las terapias de rehabilitación incrementa los beneficios respecto a las terapias manuales al incorporar tareas de ejercicio intensivas e interactivas [3,4]. Es importante mencionar que los sistemas robóticos utilizados para asistencia en rehabilitación no tienen la finalidad de reemplazar al terapeuta, sino que constituyen una valiosa herramienta que potencializa su labor.

Aunque existan sistemas automáticos para asistencia en rehabilitación física desarrollados en otros países, cuando estos llegan a México los costos de los equipos los hacen inaccesibles para la mayoría de las instituciones médicas del sector público. En esta línea, el único dispositivo para rehabilitación de dedos, actualmente disponible de manera comercial, es el desarrollado por la empresa austriaca Tyromotion [5], denominado como Amadeo®, que permite la rehabilitación de los cinco dedos; sin embargo, su precio se encuentra aproximadamente en 43,000 euros [5], lo que lo hace aun de difícil acceso para instituciones médicas públicas.

Los equipos especializados para la rehabilitación física de personas resultan ser una rama de desarrollo poco explorada, y que necesita ser cubierta para responder a la necesidad de la población afectada, y que no cuenta con los recursos económicos para adquirir un equipo comercial de rehabilitación.

[1] Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada, Instituto Politécnico Nacional, Querétaro, Qro. México.
[2] División de Tecnologías de Automatización e Información, Universidad Tecnológica de Querétaro, Querétaro, Qro. México.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

En México, en cifras oficiales del 2010, el Instituto Nacional de Estadística y Geografía (INEGI), reportó que el 5.1% de la población resultó con alguna discapacidad, y que el 58.3% de ellos tiene alguna limitación para caminar o moverse [6]. En este trabajo, presentamos el desarrollo y validación de un dispositivo robótico coadyuvante para la rehabilitación de dedos de la mano, el cual tiene por objetivo realizar movimientos asistidos de flexo-extensión de los dedos.

II. DISEÑO DE DISPOSITIVO ROBÓTICO

Se desarrolló un dispositivo robótico de asistencia para rehabilitación de dedos de la mano, que es reconfigurable y puede ser utilizado en movilización pasiva y activa de flexión y extensión de los dedos de la mano, como ejercicio de las articulaciones en mecanoterapias de rehabilitación física.

El sistema está compuesto de una base fija de altura variable que sujeta el antebrazo, cuatro mecanismos RRRT tipo manivela–biela–corredera, reconfigurables al tamaño de los dedos de la mano, impulsados por actuadores rotacionales ubicados en los cuatro planos de flexo-extensión de los dedos índice, medio, anular y meñique, ver Figura 2.1.

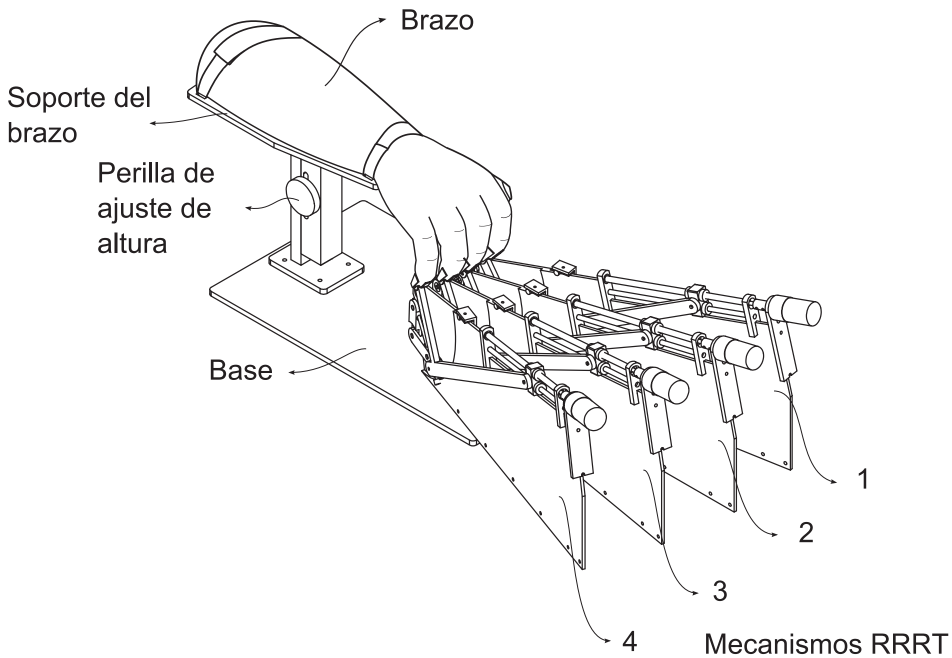


Figura 2.1. Dispositivo robótico para rehabilitación de dedos de la mano.

Los mecanismos RRRT reconfigurables movilizan las articulaciones metacarpofalángica e interfalángicas proximal y distal de los dedos índice a meñique, guiando el extremo distal de cada dedo por la trayectoria natural de flexoextensión, lo que evita el movimiento del dedo fuera de su espacio de trabajo, reduciendo el riesgo de lesión por hiperextensión. Los ejercicios de movilización del extremo distal del dedo pueden ser pasivos, es decir, que se proporciona la fuerza de movilización por el mecanismo, o activos, donde el usuario aporta la fuerza que produce el movimiento.

En virtud de alcanzar la eficiencia del dispositivo se desean movimientos suaves y continuos desde el inicio ($\boldsymbol{Lim_{flex}}$) hasta el fin del recorrido ($\boldsymbol{Lim_{ext}}$). Por lo anterior, la amplitud y velocidad de los movimientos deben controlarse para cada instante de tiempo. Se puede comenzar con oscilaciones de baja amplitud y velocidad e incrementar ambas hasta alcanzar el intervalo completo de flexo-extensión, siempre bajo estricta supervisión de fisioterapeutas profesionales para evitar movimientos bruscos que causen excesivo dolor o daño adicional.

La longitud de cada dedo define las dimensiones del mecanismo correspondiente. La síntesis de los mecanismos flexo-extensores se realizó a partir de un método de generación de trayectoria con tres puntos de precisión con base en las curvas del eslabón acoplador [7], ubicado en la Figura 2.2 por el punto ***C***.

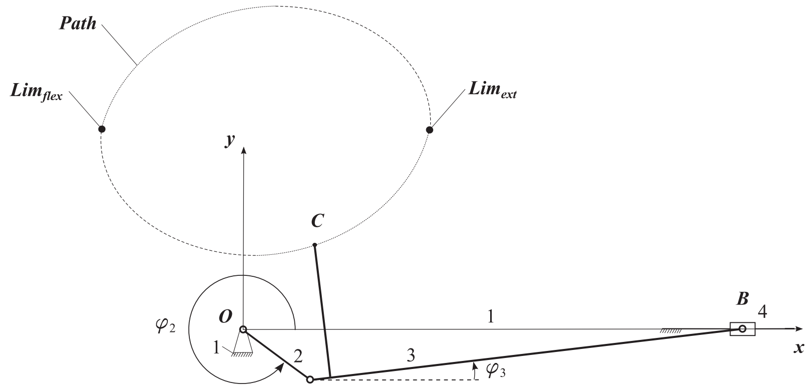


Figura 2.2. Mecanismo RRRT representativo del dispositivo robótico.

III. ANÁLISIS CINEMÁTICO DEL DISPOSITIVO

El análisis cinemático del mecanismo se realizó a través del análisis vectorial, con el cual se determina la posición, velocidad y aceleración del punto de unión entre el extremo móvil del dedo y el extremo del eslabón acoplador, punto ***C***. En la Figura 3.1 se muestra el diagrama vectorial que representa al mecanismo RRRT; el vector $\boldsymbol{r}_A$ tiene su punto inicial en el origen ***O*** del marco de referencia local ***xy***, y su punto final en la unión con el eslabón acoplador, el punto ***A***. El vector $\boldsymbol{r}_B$ también tiene su punto inicial en el origen ***O*** y su punto final en ***B***, el cual se desplaza horizontalmente sobre el eje ***x***. El vector $\boldsymbol{r}_{AC}$ tiene su origen en el punto ***A*** y su punto final en el extremo del eslabón acoplador que se une al extremo del dedo a movilizar, el punto ***C*** y el vector $\boldsymbol{r}_C$ también tiene su punto inicial en el origen ***O*** y su punto final en ***C***. Finalmente, el vector $\boldsymbol{r}_{AB}$ tiene su punto inicial en ***A*** y su punto final en ***B*** [8].

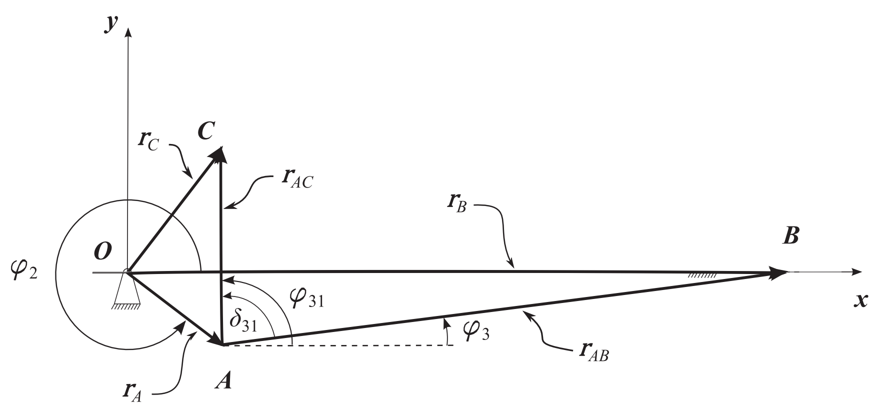


Figura 3.1. Diagrama vectorial del mecanismo RRRT.

Por lo anterior, se establece que:

$$\boldsymbol{r}_B = \boldsymbol{r}_A + \boldsymbol{r}_{AB} \tag{3.1}$$

$$\boldsymbol{r}_C = \boldsymbol{r}_A + \boldsymbol{r}_{AC} \tag{3.2}$$

El punto ***A*** gira respecto del origen ***O*** en función del ángulo φ, por lo que La posición, velocidad y aceleración del punto ***A*** está definida por el vector $\boldsymbol{r}_A$ y su primera y segunda derivadas respecto del tiempo [8]:

$$\boldsymbol{r}_A = x_A\hat{\imath} + y_A\hat{\jmath} = L_2 \cos\varphi_2\, \hat{\imath} + L_2 \operatorname{sen}\varphi_2\, \hat{\jmath} \tag{3.3}$$

$$\boldsymbol{v}_A = \dot{\boldsymbol{r}}_A = \dot{x}_A\, \hat{\imath} + \dot{y}_A\hat{\jmath} = -L_2\dot{\varphi}_2 \operatorname{sen}\varphi_2\, \hat{\imath} + L_2\dot{\varphi}_2 \cos\varphi_2\, \hat{\jmath} \tag{3.4}$$

$$\boldsymbol{a}_A = \ddot{\boldsymbol{r}}_A = \ddot{x}_A\, \hat{\imath} + \ddot{y}_A\, \hat{\jmath} = \tag{3.5}$$

$$= (-L_2\ddot{\varphi}_2\, sen\, \varphi_2 - L_2\dot{\varphi}_2^2\, cos\, \varphi_2)\hat{\imath} + (L_2\ddot{\varphi}_2\, cos\, \varphi_2 - L_2\dot{\varphi}_2^2\, sen\, \varphi_2)\hat{\jmath}$$

donde L_2 es la magnitud del vector $\boldsymbol{r}_A$, $\dot{\varphi}_2\hat{k}$ es la velocidad angular ω_2 y $\ddot{\varphi}_2\hat{k}$ es la aceleración angular α_2 [8].

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

El punto ***B*** únicamente se desplaza sobre el eje ***x,*** por lo que su componente en el eje ***y*** es constante y de magnitud nula. La posición, velocidad y aceleración del punto ***B*** está definida por el vector $\boldsymbol{r}_B$ y su primera y segunda derivadas respecto del tiempo [8]:

$$(x_B - x_A)^2 + (y_B - y_A)^2 = {L_3}^2 \tag{3.6}$$

$$\boldsymbol{r}_B = \boldsymbol{r}_A + \boldsymbol{r}_{AB} = x_B\hat{\imath} + y_B\,\hat{\jmath} = \left(x_A + \sqrt{{L_3}^2 - y_A^2}\right)\hat{\imath} + \ 0\hat{\jmath} \tag{3.7}$$

$$\boldsymbol{v}_B = \ \dot{\boldsymbol{r}}_B = \ \dot{x}_B\,\hat{\imath} + \dot{y}_B\hat{\jmath} \ = (-L_2\dot{\varphi}_2 \operatorname{sen}\varphi_2 + L_2\,k_3 \operatorname{sen}\varphi_2)\hat{\imath} + 0\hat{\jmath} \tag{3.8}$$

$$\boldsymbol{a}_B = \ \ddot{\boldsymbol{r}}_B = \ \ddot{x}_B\,\hat{\imath} + \ddot{y}_B\,\hat{\jmath} \ = \left(-\ddot{\varphi}_2 L_2 \operatorname{sen}\varphi_2 - L_2\dot{\varphi}_2^2\, cos\,\varphi_2 + \ k_2\,L_2 \operatorname{sen}\varphi_2 - k_3^2\sqrt{{L_3}^2 - y_A^2}\right)\hat{\imath} + \ 0\hat{\jmath} \tag{3.9}$$

con:

$$k_3 = \frac{-L_2\dot{\varphi}_2\, cos\,\varphi_2}{\sqrt{{L_3}^2 - y_A^2}}$$

y

$$k_2 \ = \frac{-L_2\ddot{\varphi}_2\, cos\,\varphi_2 + L_2\dot{\varphi}_2^2 \operatorname{sen}\varphi_2 - L_2 k_3^2 \operatorname{sen}\varphi_2}{\sqrt{{L_3}^2 - y_A^2}}$$

donde L_3 es la magnitud del vector $\boldsymbol{r}_{AB}$ que es constante [8].

El punto ***C*** es el extremo del eslabón acoplador que define la trayectoria deseada del extremo móvil del dedo. El ángulo φ_3 se forma entre el vector $\boldsymbol{r}_{AB}$ y la línea que pasa por el punto ***A,*** paralela al eje ***x*** y está definido por (3.10) [8]:

$$\varphi_3 = \ arco\ seno\left(\frac{-y_A}{L_3}\right) = arco\ seno\left(\frac{-L_2 seno\ \varphi_2}{L_3}\right) = \frac{-L_2}{L_3}\varphi_2 \tag{3.10}$$

El ángulo del vector $\boldsymbol{r}_{AC}$ respecto del eje horizontal es [8]:

$$\varphi_{31} = \varphi_3 + \delta_{31} \tag{3.11}$$

La posición, velocidad y aceleración del punto ***C*** está definida por el vector $\boldsymbol{r}_C$ y su primera y segunda derivadas respecto del tiempo [8]:

$$\boldsymbol{r}_C = \boldsymbol{r}_A + \boldsymbol{r}_{AC} = (x_A + \ x_{AC})\hat{\imath} + (y_A \ + y_{AC}\,)\hat{\jmath} \tag{3.12}$$

$$= (L_2 \cos\varphi_2 + \ L_{31}\cos\varphi_{31})\hat{\imath} + (L_2 \operatorname{sen}\varphi_2 + L_{31} \operatorname{sen}\varphi_{31})\hat{\jmath}$$

$$\boldsymbol{v}_C = \ \dot{\boldsymbol{r}}_C = \dot{\boldsymbol{r}}_A + \dot{\boldsymbol{r}}_{AC} = \ \dot{x}_C\,\hat{\imath} + \dot{y}_C\,\hat{\jmath} \ = \tag{3.13}$$

$$= -(L_2\dot{\varphi}_2 \sin\varphi_2 + \ L_{31}\dot{\varphi}_{31} \sin\varphi_{31})\hat{\imath} + (L_2\dot{\varphi}_2\cos\varphi_2 + L_{31}\dot{\varphi}_{31}\cos\varphi_{31})\hat{\jmath}$$

$$\boldsymbol{a}_C = \ \ddot{\boldsymbol{r}}_C = \ddot{\boldsymbol{r}}_A + \ddot{\boldsymbol{r}}_{AC} = \ \ddot{x}_C\,\hat{\imath} + \ddot{y}_C\,\hat{\jmath} \ = \tag{3.14}$$

$$\begin{aligned} = (-L_2\ddot{\varphi}_2\, sin\,\varphi_2 - L_2\dot{\varphi}_2^2\, cos\,\varphi_2 - L_{31}\ddot{\varphi}_{31}\, sen\,\varphi_{31} - L_{31}\dot{\varphi}_{31}^2\, cos\,\varphi_{31})\hat{\imath} \\ + (L_2\ddot{\varphi}_2\, cos\,\varphi_2 - L_2\dot{\varphi}_2^2\, sen\,\varphi_2 + L_{31}\ddot{\varphi}_{31}\, cos\,\varphi_{31} \\ - L_{31}\dot{\varphi}_{31}^2\, sin\,\varphi_{31})\hat{\jmath} \end{aligned}$$

donde L_{31} es la magnitud del vector $\boldsymbol{r}_{AC}$ [8].

Las Figuras 3.2 a 3.4 muestran las trayectorias de los puntos ***A***, ***B*** y ***C*** del mecanismo propuesto cuando la variable de entrada es el desplazamiento en ***B***.

Robótica y Mecatrónica de Servicios: Teoría y Aplicaciones. Capítulo 22, páginas 211 - 220.
ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

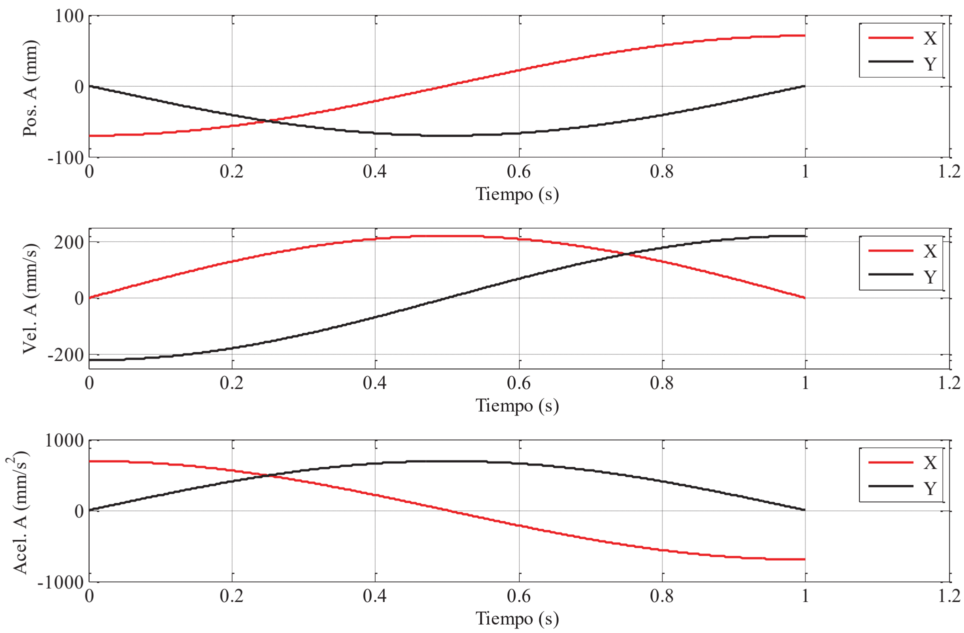


*Figura 3.2. Curvas de Posición, Velocidad y Aceleración del Punto **A**.*

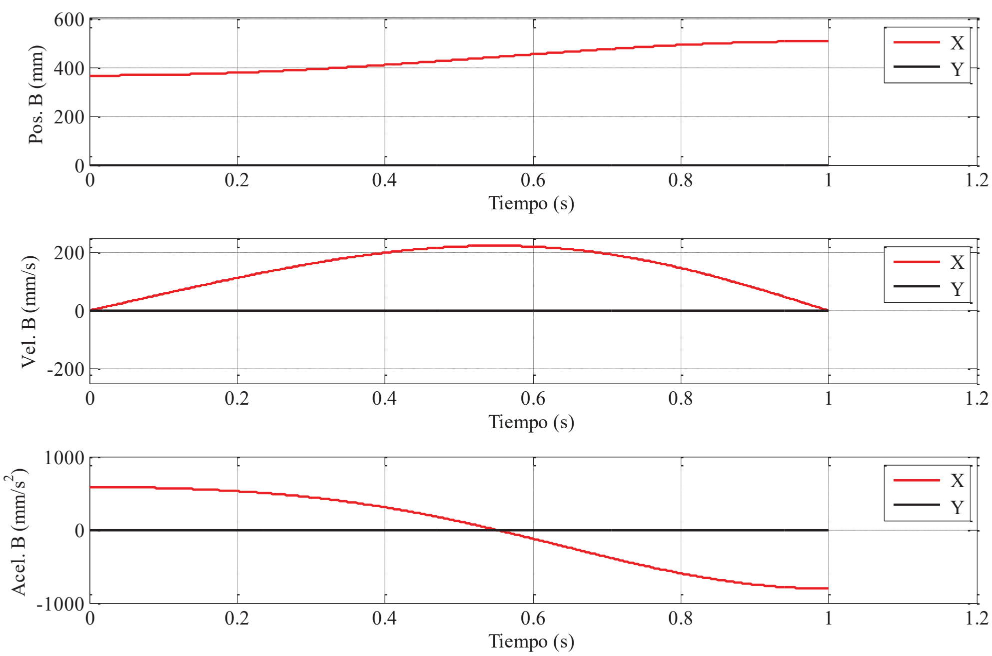


*Figura 3.3. Curvas de Posición, Velocidad y Aceleración del Punto **B**.*

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

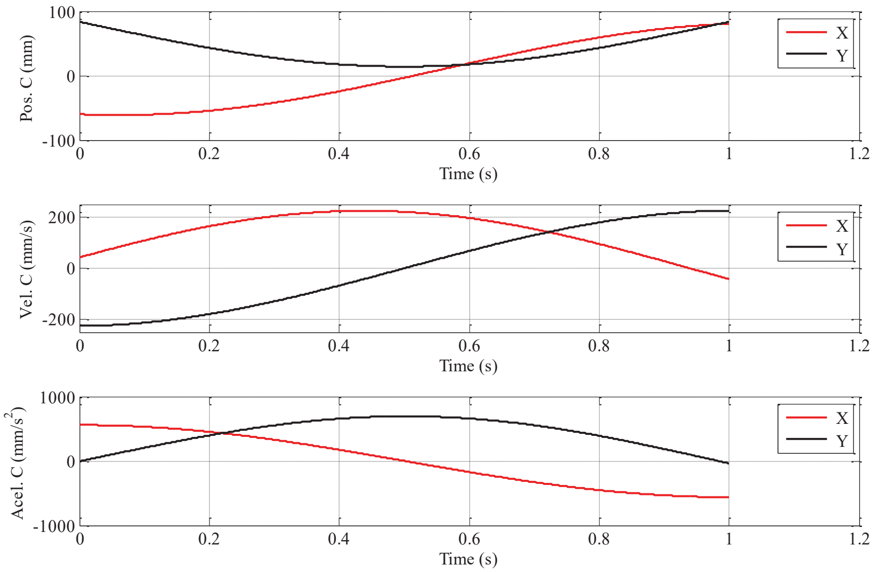


*Figura 3.4. Curvas de Posición, Velocidad y Aceleración del Punto **C**, para 180° ≤ φ ≤ 360°.*

IV. VALIDACIÓN EXPERIMENTAL

Para la validación del dispositivo se realizó el análisis de trayectoria que este genera en su efector final, punto ***C***, donde se posiciona el dedo del paciente. En la Figura 4.1 se muestra el dibujo del mecanismo propuesto para un solo dedo.

Como primer paso en la validación experimental se realizó el control del mecanismo para generar las trayectorias de flexión y extensión.

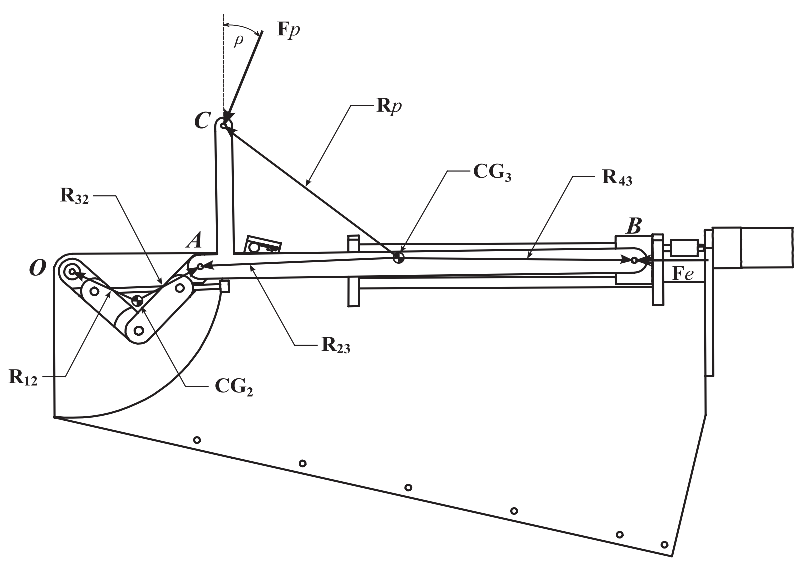


Figura 4.1. Mecanismo de cuatro barras reconfigurable para un dedo.

Debido a que los actuadores que producen el movimiento de los mecanismos flexoextensores son servomotores de corriente directa, se seleccionó el sistema de control PicServo®, ver Figura 4.2. Este sistema comercial consiste en una tarjeta de comunicación SSA-485 Smart Serial Adapter y cuatro tarjetas controladoras de movimiento PIC-SERVO SC Motion Control Board.

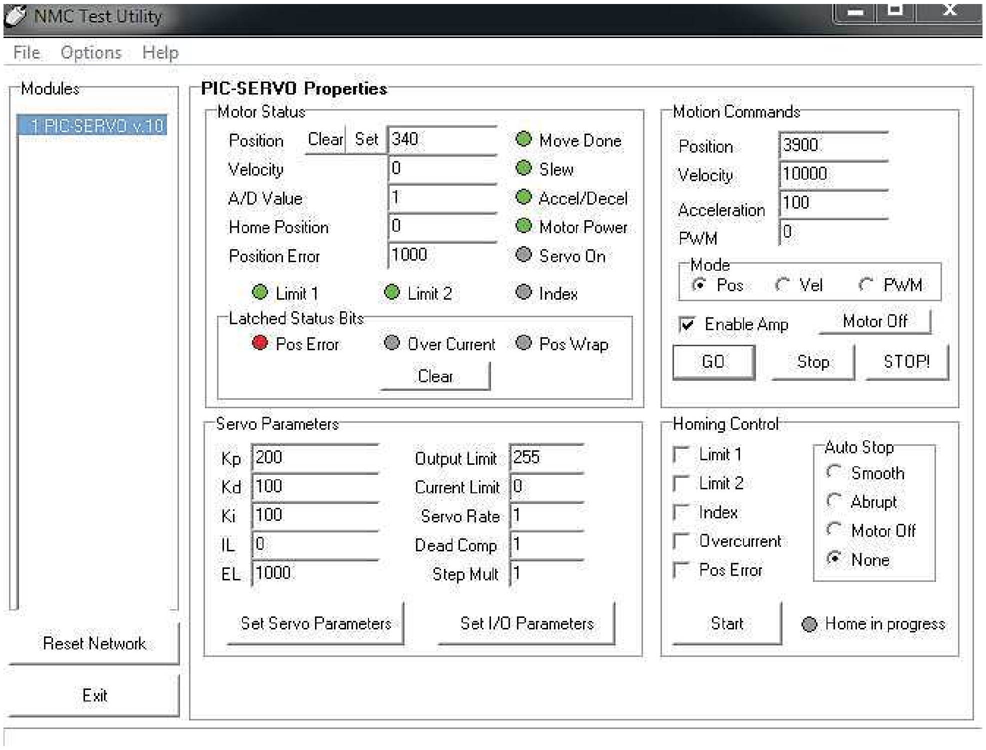


Figura 4.2. Interfaz PicServo®.

Se instalaron interruptores de fin de carrera en los extremos de la trayectoria de la corredera para limitar su desplazamiento.

Posterior a generar los movimientos controlados de flexión y extensión en el mecanismo, se procedió a realizar tomas de videos correspondientes a las trayectorias, y mediante software, la descomposición de éstos en imágenes; para luego analizar cada una de ellas y detectar el efector final, ver Figura 4.3.

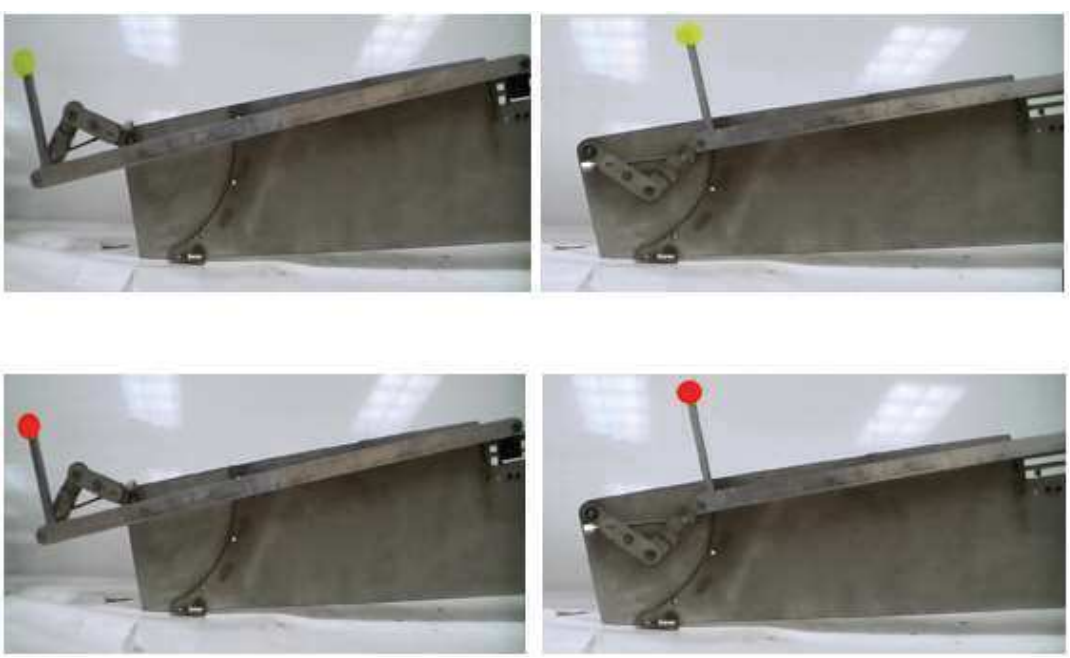


*Figura 4.3. Trayectoria del punto **C**.*

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

Con el punto ***C*** identificado, correspondiente al efector final, en cada una de las imágenes, se realizó la clasificación de este punto respecto de los demás datos de la imagen; se le asignó el color negro al punto ***C*** y el color blanco a los demás datos de la imagen, tal como se aprecia en la Figura 4.4.



*Figura 4.4. Clasificación del punto **C** (color negro) respecto de los demás datos de la imagen.*

Una vez realizada dicha asignación, se analizó el centro de masa del punto del punto ***C*** mediante coordenadas ***xy***, tal cual se muestra en la Figura 4.5.

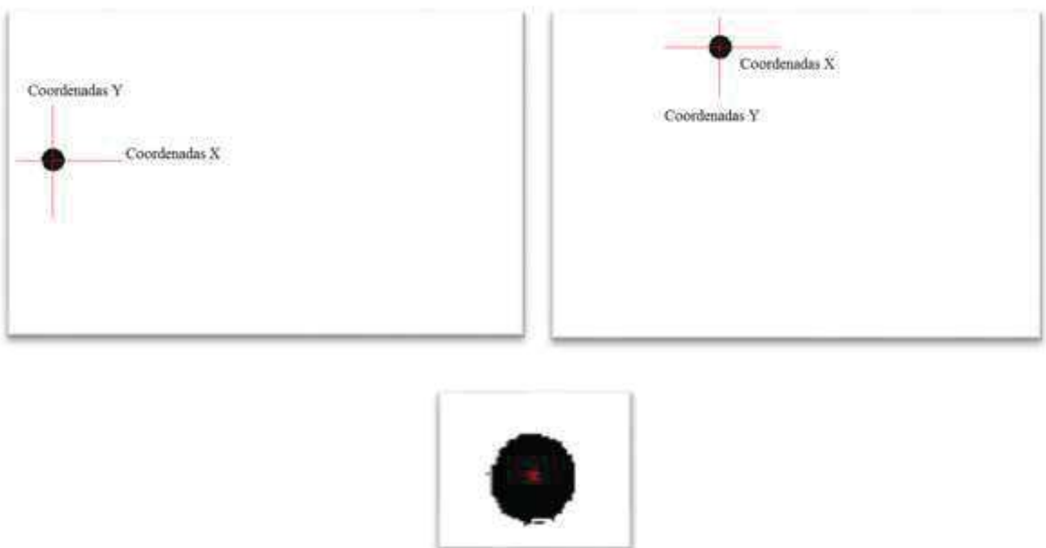


*Figura 4.5. Centro de masa del punto del punto **C**.*

Con las coordenadas ***xy*** de cada imagen se procedió a graficar los puntos para visualizar la trayectoria que el mecanismo genera de los movimientos de flexión y extensión.

En la Figura 4.6 se ilustran las trayectorias del mecanismo, de color negro se observa la trayectoria de los movimientos de flexión a extensión, y en color rojo la trayectoria de extensión a flexión. Nótese que las trayectorias que genera el mecanismo son coincidentes a lo largo de los movimientos de flexión y extensión, sin embargo, en su punto inicial de flexión, el mecanismo presenta juego mecánico debido a la estructura de la articulación.

Finalmente se analizaron las posiciones en las coordenadas ***xy*** del mecanismo respecto al tiempo de ejecución de los movimientos de flexión y extensión. En la Figura 4.7(a) se observa la posición de la coordenada ***x*** de color negro, y de color rojo la posición de la coordenada ***y***; estas coordenadas se analizaron respecto al tiempo de ejecución del movimiento de flexión-extensión.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

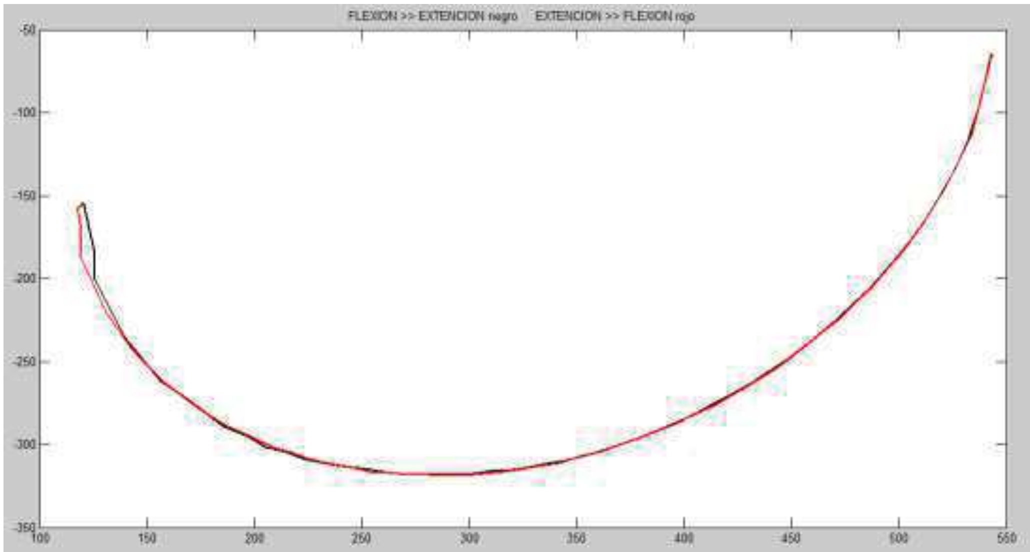
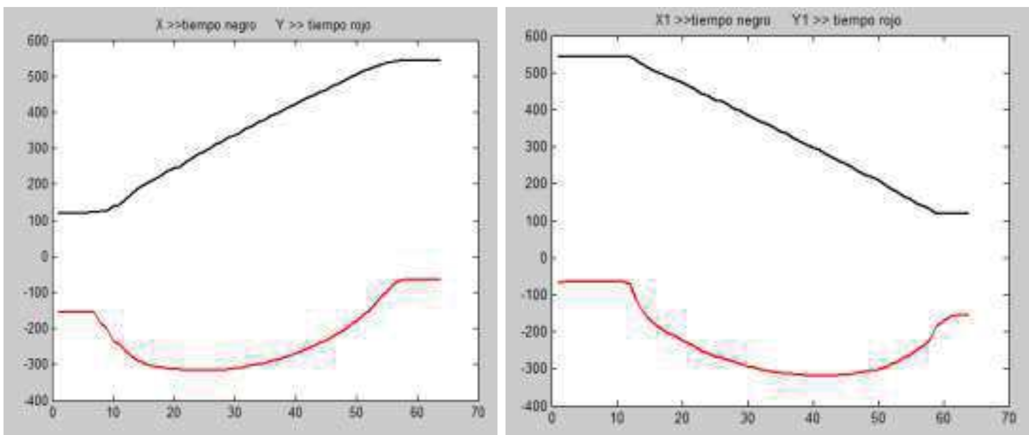


Figura 4.6. Trayectorias del mecanismo: movimientos de flexión a extensión (color negro) y movimientos de extensión a flexión (color rojo).



(a) *(b)*

*Figura 4.7. Coordenadas **xy** respecto al tiempo de ejecución del movimiento de: (a) flexión-extensión, (b) extensión- flexión.*

En la Figura 4.7(b) se aprecia la posición de la coordenada ***x*** de color negro, y de color rojo la posición de la coordenada ***y***, estas coordenadas se analizaron respecto al tiempo de ejecución del movimiento de extensión-flexión.

Nótese que la posición inicial del movimiento de flexión–extensión en las coordenadas ***xy***, es la posición final del movimiento extensión–flexión, y que la posición final del movimiento flexión–extensión en coordenadas ***xy***, es la posición inicial del movimiento de extensión–flexión.

En conclusión, el mecanismo genera dos trayectorias similares, una para flexión–extensión y otra para extensión–flexión. Las cuales representan la posición del efector final.

V. TRABAJO FUTURO

Como parte del trabajo próximo a desarrollarse se planea diseñar y habilitar el mecanismo para la rehabilitación del dedo pulgar, que por sus características anatómicas difiere de los movimientos de los demás dedos de la mano. Esto con la finalidad de obtener un prototipo para los cinco dedos, el cual deberá ser evaluado para determinar la eficiencia del mismo y la forma de control más apropiada.

También es deseable obtener los valores correspondientes a las fuerzas resultantes del análisis dinámico del mecanismo, así como sus pertinentes pruebas con pacientes, bajo supervisión médica y con la aprobación de los comités de ética pertinentes.

ISBN 978-607-9394-06-6, Asociación Mexicana de Mecatrónica, A. C.

VI. CONCLUSIONES

Bajo el esquema de contribuir en el aumento en la calidad de las terapias de rehabilitación de dedos de la mano, se desarrolló un dispositivo robótico reconfigurable que desarrolla los movimientos naturales de flexo-extensión de los dedos de la mano, exceptuando el pulgar. El dispositivo se puede ajustar a distintos tamaños de dedos de la mano, ampliando el rango de atención para pacientes que requieran rehabilitación de dedos.

Se realizó el correspondiente análisis cinemático del mecanismo propuesto, así como la validación de las trayectorias correspondientes, obteniendo resultados que corroboran la correcta ejecución de la tarea a desempeñar. A través de la validación del mecanismo se comprueba que las trayectorias correspondientes a los movimientos de flexión–extensión y extensión–flexión son semejantes.

Trabajamos para que, a mediano plazo, el prototipo cuente con la inclusión de los movimientos para la rehabilitación del dedo pulgar y la mejora respectiva en el control del dispositivo. Dejándolo habilitado para las pruebas pertinentes con pacientes.

Es importante enfatizar que el sistema debe ser usado bajo estricta supervisión de un fisioterapeuta o personal médico capacitado.

REFERENCIAS

[1] Silva, C. A., Muñoz, J. E., Garzón, D. A., Landínez, N. S., & Silva, O. (2011). Diseño mecánico y cosmético de una prótesis parcial de mano. *Revista Cubana De Investigaciones Biomédicas, 30*(1), 15-41. Recuperado de http://scielo.sld.cu/pdf/ibi/v30n1/ibi03111.pdf

[2] Gorgues, J. (2009, Febrero). Ortesis para los dedos de la mano. *Offarm*, 28(2), 68-71. Recuperado de http://apps.elsevier.es/watermark/ctl_servlet?_f=10&pident_articulo=13132753&pident_usuario=0&pcontactid=&pident_revista=4&ty=124&accion=L&origen=zonadelectura&web=www.elsevier.es&lan=es&fichero=4v28n02a13132753pdf001.pdf

[3] Burgar, C. G., Lum, P. S., Shor, P. C., & Machiel Van der Loos, H. F. (2000). Development of robots for rehabilitation therapy: the Palo Alto VA/Stanford experience. *Journal of Rehabilitation Research and Development*, *37*(6), 663–73. Recuperado de http://www.ncbi.nlm.nih.gov/pubmed/11321002

[4] Heo, P., Gu, G. M., Lee, S., Rhee, K., & Kim, J. (2012). Current hand exoskeleton technologies for rehabilitation and assistive engineering. *International Journal of Precision Engineering and Manufacturing*, *13*(5), 807–824. Recuperado de http://doi.org/10.1007/s12541-012-0107-2

[5] González, M. (2014, Noviembre 30). Evaluabilidad de los programas del robot Amadeo en la rehabilitación de la mano del hemipléjico. *TOG (A Coruña), 11*(20), 1-20. Recuperado de http://www.revistatog.com/num20/pdfs/original9.pdf

[6] INEGI. (2010). *Las personas con discapacidad en México , una visión al 2010*. (Instituto Nacional de Estadística y Geografía (México)., Ed.) (2010th ed.). México.

[7] Norton, R. (2012). *Design of Machinery: An Introduction to the Syntesis and Analysis of Mechanisms and Machines*. (MacGraw Hill, Ed.) (5a. ed.).

[8] Aguilar, J. F. (2016). *Sistema robótico para asistencia en terapias de rehabilitación de movimientos de la mano* (Tesis Doctoral, Instituto Politécnico Nacional, 2016) (pp. 1-101). Querétaro.

Sobre los autores

E. Castillo Castañeda

Doctor en Control Automático por el Instituto Nacional Politécnico de Grenoble, Francia (1994). Ingeniero Mecánico Electricista por Universidad Nacional Autónoma de México. Trabajó en el INRIA (Instituto Nacional de Investigación en Control Automático e Informática), en Francia. Ha sido profesor investigador en el ITESM-Campus Querétaro y en la Universidad Autónoma de Querétaro. Actualmente, es profesor de tiempo completo en el Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada, Unidad Querétaro del IPN. Es miembro del Comité de Robótica de IFTOMM (International Federation for the Promotion of Mechanism and Machine Science).

P.A. Niño Suárez

Maestra en Ingeniería Eléctrica, área Biomédica por la Universidad de Los Andes, Bogotá, Colombia, (1997). Ingeniera Electrónica de la Universidad Antonio Nariño, Bogotá, Colombia (1995). Actualmente está adscrita a la Escuela Superior de Ingeniería Mecánica y Eléctrica, Unidad Azcapotzalco y es Coordinadora del Doctorado en Ingeniería de Sistemas Robóticos y Mecatrónicos del Instituto Politécnico Nacional. Su área de investigación es el Diseño de Sistemas Mecatrónicos para aplicaciones biomédicas y de rehabilitación.

E. Morales Sánchez

Doctor en Ingeniería en el área de propiedades eléctricas de materiales y Maestría en Instrumentación y Control por la Universidad Autónoma de Querétaro. Ingeniero en Electrónica por la Benemérita Universidad Autónoma de Puebla. Ha sido profesor investigador del CICATA-IPN Unidad Querétaro desde 2006. Investigador Nacional Nivel II del CONACYT. Sus áreas de investigación son: Tecnologías emergentes para procesamiento de alimentos; Desarrollo de dispositivos mecatrónicos para el procesamiento de materiales orgánicos e inorgánicos, y Desarrollo de dispositivos para Bioelectrónica, Robótica, Electrónica.

J. E. Vargas Soto

Posdoctorado por la Universidad de Telecomunicaciones de Tokio, Japón. Doctor en Ciencias Físicas con especialidad en Informática y Automática por la Universidad Complutense de Madrid, Maestría en Tecnologías de la Información en Fabricación por la Universidad Politécnica de Madrid, Licenciatura en Ingeniería Mecánica y Eléctrica por la Facultad de Ingeniería de la Universidad Nacional Autónoma de México. Actualmente es profesor titular de tiempo completo en la Facultad de Ingeniería de la Universidad Autónoma de Querétaro. Senior member IEEE, fundador de la Asociación Mexicana de Mecatrónica A.C., Investigador Nacional Nivel 1 del CONACYT. Su áreas de investigación son en torno al desarrollo de máquinas mecatrónica.

J.M. Ramos Arreguín

Doctorado en Ciencias y Tecnología con especialidad en Mecatrónica (2008), Maestría en Ingeniería Eléctrica en la Facultad de Ingeniería Mecánica, Eléctrica y Electrónica (FIMEE). Ingeniero en Electrónica egresado de la FIMEE. Actualmente adscrito como profesor titular de tiempo completo en la Universidad Autónoma de Querétaro, Facultad de Ingeniería. Investigador Nacional Nivel 1 del CONACYT. Sus áreas de investigación comprenden diseño y desarrollo de sistemas embebidos y manipuladores neumáticos.

ISBN: 978-607-9394-06-6
9 786079 394066